

Universitat de Lleida
Escola Politècnica Superior
Enginyeria en Informàtica

Treball final de carrera

Aplicació Web de Simulacions per l'Estudi Ramader Porcí

Autors: Lluís Harguindey Cascón
Xavier Pla Moran

Director: Lluís Miquel Pla Aragonés

Març 2008

Per tots aquells que ens han donat suport
durant aquests mesos difícils. Familiars i amics, gràcies.

Índex

Capítol 0	10
Objectius Inicials	10
Capítol 1	11
Introducció	11
Capítol 2	13
Tecnologies Utilitzades	13
2.1 MySQL	13
2.2 La llibreria Jfreechart	18
2.2.1 Orígens i Requeriments.....	18
2.2.2 Característiques	18
2.2.3 Exemple pràctic de JFreeChart sobre l'aplicació realitzada	20
2.3 Java	21
2.4 Apache Tomcat.....	24
2.5 La Tecnologia Java Server Faces (JSF)	26
2.5.1 Model Vista Controlador (MVC).....	29
2.5.2 Beneficis de la Tecnologia JavaServer Faces	29
2.5.3 Versions de JSF	30
Capítol 3	31
Aplicació Realitzada	31
3.1 Arquitectura	31
3.1.1 BBDD (distribuïdes vs centralitzades)	31
3.1.1.1 Les bases de dades Centralitzades (BDC).....	32
3.1.1.2 Les bases de dades Distribuïda (BDD)	33

3.1.2 Model Client/Servidor	35
3.1.2.1 Avantatges i versatilitat del model Client/Servidor	36
3.1.3 Multiplataforma	37
3.2 Capes	39
3.2.1 Domini	40
3.2.2 Interfície	42
3.2.2.1 Interfície Swing	43
3.2.2.2 Interfície JSF	46
3.3 L10n	49
3.4 Diagrames	55
3.4.1 Diagrama Cas d'ús	55
3.4.2 Diagrama de Classes	57
3.5 Descripció i Format del fitxer VSI	59
3.6 Implementació	63
3.6.1 Aplicació Alta/Baixa d'usuaris del servidor de BD	63
3.6.2 Aplicació ASERP (Paquet Domini)	65
3.7 Funcionalitats	68
3.7.1 Aplicació de gestió d'usuaris	68
3.7.2 Funcionalitats compartides de l'aplicació web i de l'aplicació d'escriptori ...	69
3.8 Proves Realitzades	82
3.8.1 Configuracions	82
3.8.2 Marc de proves, resultats i temps d'execució	84
Capítol 4	85
Objectius Complerts i Treball Futur	85
Capítol 5	86
Conclusió	86
Capítol 6	87

Llicència	87
Acrònims	88
Glossari	89
Bibliografia.....	93
Annexos:	95
Annex A.....	96
Taules SQL.....	96
Annex B.....	99
Documentació de l'ASERP	99
Annex C	151
IDE Netbeans.....	151
Annex D	152
Llicència GPL	152

Índex Taules

Taula 1: Raó de la baixa del fitxer VSI	61
Taula 2: Càlculs d'ASERP que retornen una mitja	73
Taula 3: Càlculs d'ASERP que retornen un resultat total.....	74
Taula 4: Requeriments del Software realitzat.	82
Taula 5: Proves d'ASERP en un entorn Linux	84
Taula 6: Proves d'ASERP en un entorn windows.....	84

Índex Figures

Figura 1: Rendiment MySQL vs Oracle	15
Figura 2: Arquitectura MySQL	16
Figura 3: Exemple de gràfic generat amb JFreeChart.....	19
Figura 4: Gràfic de barres3D realitzat a l'aplicació de simulació	20
Figura 5: Dinamisme de Java en la Xarxa	22
Figura 6: Model de funcionament de JSF	27
Figura 7: Cicle de vida d'un JSF	28
Figura 8: Sistema Centralitzat.....	32
Figura 9: Sistema Centralitzat.....	32
Figura 10: Arquitectura BDD amb servidors de BD en diferent localització.....	33
Figura 11: Arquitectura BDD amb servidors de BD en la mateixa localització.	34
Figura 12: Peticions client/servidor aplicació no web	35
Figura 13: Peticions client/servidor aplicació web.....	36
Figura 14: Exemple interacció entre diferents S.O.....	38
Figura 15: Representació de la programació per capes	39
Figura 16: Esquema funcionament aplicació per capes	41
Figura 17: Exemple botó Swing	45
Figura 18: Exemple botó JSF.....	47
Figura 19: Codi del fitxer JSP	48
Figura 20: Codi del fitxer Java	48
Figura 21: Fragment de codi del fitxer CalculsBase.java.....	52
Figura 22: Fragment de codi del fitxer index.java	53
Figura 23: Alternativa als fitxers properties.....	54

Figura 24: Procés d'alta d'un usuari al servidor de BD.....	64
Figura 25: Procés d'alta d'un usuari al servidor de BD.....	64
Figura 26: Arquitectura de la funcionalitat del l'ASERP.	65
Figura 27: Pantalla d'entrada de l'administrador d'usuaris.....	68
Figura 28: Pantalla d'alta/baixa d'usuaris	69
Figura 29: Pantalla d'autenticació d'usuaris de l'aplicació web.....	70
Figura 30: Pantalla de menú principal de l'aplicació web.....	70
Figura 31: Exemple pujar fitxer VSI	71
Figura 32: Acotació paràmetres per realitzar càlculs.....	72
Figura 33: Exemple càlculs numèrics	75
Figura 34: Exemple gràfic per interval	76
Figura 35: Exemple gràfic per cicle	77
Figura 36: Exemple gràfic per dia.....	78
Figura 37: Exemple gràfic per setmana.....	79
Figura 38: Consulta genèrica sobre la taula altas i resultat d'aquesta	80
Figura 39: Pantalla de selecció d'idioma de l'aplicació web	81
Figura 40: Codi inicialització de la BD.....	83

Capítol 0

Objectius Inicials

- Estudi de les dades rellevants en la reproducció d'animals en les granges porcines.
- Anàlisi del format del fitxer subministrat (Fitxer VSI) que conté les dades procedents de granges reals.
- Disseny d'una Web per a la realització de simulacions complexes sobre les mostres de dades de granges disponibles.
- Elaboració d'un software gestor d'altres i baixes d'usuaris en una BD, aquests usuaris han de disposar d'accés a l'aplicació per poder llegir la base de dades que se'ls hi ha estat assignada en la seva alta; a fi de realitzar posteriorment consultes, càlculs numèrics o gràfics del comportament reproductiu en el temps.
- Els usuaris de l'aplicació poden tenir diferents permisos que els proporcionaran poder estudiar una o més granges. Depenent de si és un usuari amb permisos limitats o no.
- Implementació d'ASERP (Aplicació de Simulacions per l'Estudi Ramader Porcí) la qual serà una aplicació que treballa en entorn Web.

Capítol 1

Introducció

Segons estudis realitzats els últims anys les terres de Lleida, han estat el territori més tardà en implantar la ramaderia intensiva. Tot i així, és el que més ha incrementat la producció, fins a situar-se en el primer lloc en nombre d'explotacions de porcs i vedells d'engreix de Catalunya, i un dels capdavanters d'Europa. Això ha estat possible gràcies al sacrifici de molts ramaders i en la incorporació de les tecnologies en el món ramader.

Sector	Caps de bestiar	
	Segrià (Comarca)	Lleida (Població)
Porcí	715.922	2.327.064
Boví	107.523	274.779
Oví	115.818	503.805
Avícola	5.780.722	19.505.867
Canícula	59.894	512.273

Font DARP, 6/2000

Pel que fa al sector porcí, cal destacar la seva importància, tant pel que fa a la quantitat de caps com per la presència a la zona de les principals empreses del sector (Vall Companys SA, Copaga S.COOP, Corporació Alimentària Guissona SA).

És per això, que en uns dies on cada cop és més important la tecnologia de les dades i la informació, el món de les professions del sector primari no pot permetre's el luxe de viure en el passat; sinó que ha d'anar evolucionant al mateix ritme del que és rodeja.

En l'indústria porcina, com en totes les rames de la indústria i la economia del món, es produeix un augment constant de la necessitat d'automatització dels processos. La velocitat de creixement de la indústria Electrónica a nivell mundial ha arribat nivells exponencials, dotant a la resta de les indústries d'eines potents per elevar l'eficiència i eficàcia de les seves produccions.

En l'actualitat són molts els sistemes informàtics que s'han creat per l'indústria porcina amb la finalitat d'automatitzar tot el procés de cria porcina. Aquests es varen centrar en els seus inicis en l'emmagatzemament de tota la informació de les granges. Amb l'arribada de les noves tecnologies informàtiques i la revolució d'Internet, va anar augmentant el camp d'acció d'aquests sistemes i van incloure en els seus programes altres facilitats que permeten emmagatzemar informació de grups de granges. D'aquesta manera es poden comparar sistemes de producció diferents.

Actualment, en alguns països ja s'han utilitzat sistemes informàtics per a intentar incrementar la producció i reduir costos. Concretament en el nostre, s'han utilitzat alguns d'aquests sistemes en granges aïllades. Alguns d'aquests sistemes han estat per exemple PIGCHAMP d'Estats Units, ZAP provinent d'Holanda, PIGCONTROL desenvolupat en el nostre país i el WINPORC francès.

Dit tot això i veient que és un punt que s'ha de tindre molt en compte per tal de millorar l'economia i la producció del món porcí; quelcom eina per l'estudi que ens ajudi a realitzar una millor presa de decisions i alhora que aquesta ens produeixi majors beneficis, serà oportuna la seva implementació.

L'elaboració d'aquest projecte a obert una nova eina per l'estudi de la producció en granges porcines. Possiblement cal recórrer encara molt camí per a que el software realitzat es pugui ajustar a totes les exigències que pugui tenir el mercat de grangers de porcs. Però en aquestes paraules és on comença el seu camí.

Capítol 2

Tecnologies Utilitzades

A continuació passarem a anomenar i descriure els trets més significatius de les diferents tecnologies utilitzades per a desenvolupar l'ASERP (Aplicació Simulacions Estudi Ramaderia Porcina); com ha estat les bases de dades MySQL [12], el llenguatge Java, la llibreria per a realitzar gràfics JfreeChart [10], el servidor Web Apache Tomcat [0] i aquest nou llenguatge de SUN per a les interfícies Web; el format JSF.

2.1 MySQL

Un dels punts claus en el desenvolupament d'aquest projecte, va ésser l'elecció d'una de les bases de dades (BBDD) de les quals se'ns ofereix lliure o comercialment en el mercat.

Una base de dades és un conjunt de dades relacionades entre si, que es poden consultar i que cal mantenir, aquestes es troben emmagatzemades en una o més taules. Cada taula representa alguna estructura d'informació rellevant (clients d'una empresa, altes d'usuaris, catàlegs d'una biblioteca, etc.). A més de les dades, també es poden trobar relacionades les taules mitjançant elements comuns en dues o més taules diferents. Per exemple en una llibreria, podem trobar una base de dades que contingui una taula amb tots els usuaris abonats en aquella llibreria, una altra taula amb els noms dels llibres, el seu corresponent identificador i una breu descripció. A més d'una altra taula que emmagatzemi els llibres que s'han alquilat. En aquesta última taula apart d'estar relacionades les dades que el componen, també serveix de vincle entre les altres dues taules. Ja que per la taula on es tenen els lloguers, ha de constar forçosament del identificador del client que ha retirat el llibre (Taula clients) i el identificador del llibre retirat.

Altres exemples de bases de dades són el sistema de facturació d'una empresa o el sistema d'avaluació d'una institució escolar o simplement d'un professor.

És un concepte tan antic com la recollida d'informació d'un col·lectiu, una collita o una cultura. Però és amb els sistemes informàtics que aquest concepte agafa una nova dimensió.

Una base de dades forma part d'un sistema d'informació. Els sistemes d'informació computats gestionen les bases de dades a través de programes informàtics anomenats sistemes de gestió de base de dades.

Perquè siguin útils les dades s'han de poder consultar, per pantalla, obtenir informes, o mitjançant altres sistemes informàtics. També és necessari que tingui un manteniment que actualitzi les dades i corregeixi els errors detectats.

Una base de dades pot seguir un model de dades. El més comú és el model relacional de dades i el llenguatge més conegut per a aquest model de dades és el SQL.

És per aquest fet que hem decidit estudiar les possibilitats de les que disposem de models relacionals sobre llenguatge SQL.

Entre aquest sector hem trobat les bases de dades de Pilot-DB [15], PostgreSQL [16], Col·lex [2], SQL Server, Oracle i Mysql.

De tots aquests hem cregut oportú centrar-nos en els dos que actualment semblen despuntar respecte els altres. Aquests són Oracle i MySQL. Curiosament el primer és software comercial, mentre que el segon es gratuït i OpenSource [13].

MySQL utilitza una llicència sota els termes de la Llicència Pública General GNU. És gratis per a ús intern i per als ISP. Per exemple és gratuït si ho utilitza per a desenvolupament de programari o per a les companyies que ho usen com servidor de base de dades, i també és gratis si es desenvolupa una aplicació a mesura per a un client. Per a altres finalitats la llicència llavors té un valor unitari de \$200 (USD).

D'aquesta manera l'elecció de MySQL també aconseguirà abaratir els costos ja que no cal realitzar una despesa tan important com ho hagués estat l'ús de Oracle [14].

Com la nostra finalitat és l'estudi i ens hem informat de que MySQL és més eficient per a bases de dades de pocs milers de registres que el seu competidor comercial; ens hem inclinat per a la utilització de MySQL per a gestionar les granges de cadascun dels usuaris de l'aplicació.

Podem contrastar els resultats de rendiment a partir de la figura següent:

Almacenamiento	# Registros Insertados	T1(s)	T2(s)	T3(s)	T Promedio(s)	T Promedio(min)
Innodb	100	0,000	0,010	0,010	0,007	0,000
Innodb	1.000	0,030	0,040	0,030	0,033	0,001
Innodb	10.000	0,360	0,380	0,350	0,363	0,006
Innodb	1.000.000	132,750	124,415	116,020	124,395	2,073
Innodb	3.000.000	227,510	228,980	355,740	270,743	4,512
Myisam	100	0,100	0,040	0,010	0,050	0,001
Myisam	1.000	0,050	0,070	0,050	0,057	0,001
Myisam	10.000	0,500	0,430	0,450	0,460	0,008
Myisam	1.000.000	46,130	45,520	45,230	45,627	0,760
Myisam	3.000.000	144,390	142,910	143,710	143,670	2,395
Oracle	100	0,200	0,200	0,200	0,200	0,003
Oracle	1.000	0,291	0,311	0,301	0,301	0,005
Oracle	10.000	3,155	3,157	3,194	3,169	0,053
Oracle	1.000.000	364,244	333,250	340,340	345,945	5,766
Oracle	3.000.000	1046,115	1134,270	1098,345	1092,910	18,215

Figura 1: Rendiment MySQL vs Oracle

(Innodb [6] i MyLsam [11] són dos tipus d'emmagatzemament que permet realitzar MySQL. El primer treballa amb les transaccions i l'integritat referencial, mentre que Myslam no utilitza transaccions ni integritat referencial. D'aquesta manera és encara molt més ràpid).

Un punt a tenir en compte al elegir MySQL és que quelcom usuari pot obtenir el codi font. Això fa que tot software OpenSource tingui una independència del proveïdor. D'aquesta manera es pot modelar de forma que s'adapti millor a les seves necessitats. A demés tota la informació (estat, bugs, etc) és oberta també i no existeix política d'ocultament corporativa ni censura. Així doncs si alguna cosa no funciona, es sabrà ràpidament. Com a conseqüència d'aquest fet, els projectes de codi obert són més ràpids a corregir els seus errors si sorgeixen que els comercials. Amb l'argumentació anterior, finalment hem decidit que la nostra millor opció es MySQL.

Encara ara algú es pot preguntar, què és exactament MYSQL?

Mysql primerament és una base de dades. A més també es podria definir MySQL com un servidor multi fils de bases de dades de codi obert, fiable, ràpid, compacte, poderós i multi plataforma.

Les seves principals característiques són:

- El principal objectiu de MySQL és velocitat i robustesa.
- Escrit en C i C++, testat amb GCC 2.7.2.1. Usa GNU autoconf per a portabilitat.
- Clients C, C++, Java, Perl, TCL, etc
- Multiprocés, és a dir pot usar diverses CPU si aquestes estan disponibles.
- Pot treballar en diferents plataformes i S.O. distints.
- Sistema de contrasenyes i privilegis molt flexible i segur.
- Registres de longitud fixa i variable. - 16 índexs per taula, cada índex pot estar compost de 1 a 15 columnes o parts d'elles amb una longitud màxima de 127 bytes.
- Totes les columnes poden tenir valors per defecte.
- Totes les dades estan gravades en format ISO8859_1 .
- Els clients usen TCP o UNIX Socket per connectar-se al servidor. - El servidor suporta missatges d'error en diferents llengües.
- Totes les comandes tenen -help o -? Per a les ajudes.
- Diversos tipus de columnes com enters de 1, 2, 3, 4, i 8 bytes, coma flotant, doble precisió, caràcter, dates, enumerats, etc.

Mentre que la seva arquitectura s'exposa en la següent figura:

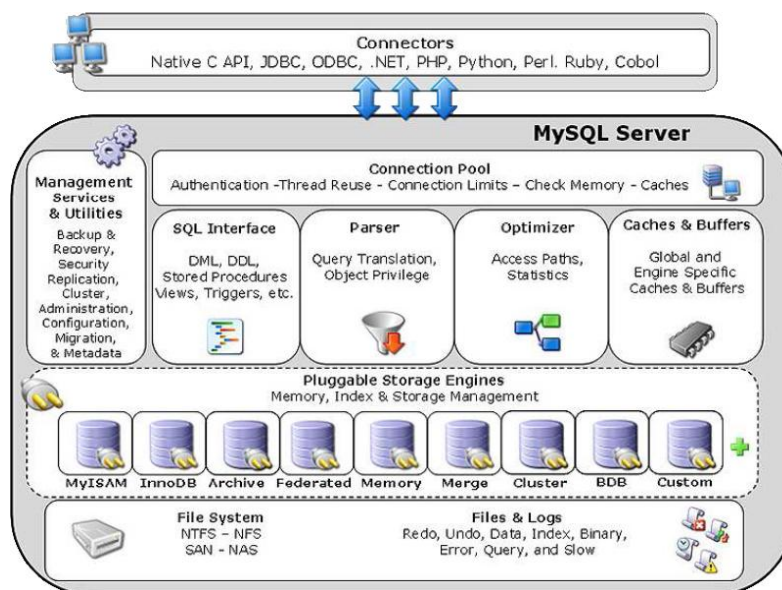


Figura 2: Arquitectura MySQL

Respecte la figura anterior no entrarem en temes d'arquitectura de l'aplicació perquè considerem que s'allunya dels nostres interessos però si hem cregut convenient mencionar que per tal de poder treballar amb una base de dades des d'una aplicació Web o d'escriptori; és necessari l'ús dels connectors.

Els connectors, són per dir-ho d'alguna manera traductors entre el llenguatge de programació i les instruccions de la base de dades. Per a dur a terme el nostre projecte hem utilitzat el JDBC.

El connector JDBC (Java Database Connectivity) de MySQL, és un conductor natiu de Java, que converteix aquest en crides en el protocol de xarxa utilitzat en les bases de dades de MySQL. El fet de treballar amb aquest ens permet utilitzar MySQL en ambients heterogenis.

2.2 La llibreria Jfreechart

JFreeChart és una llibreria gratuïta per a la plataforma de programació Java, la funcionalitat bàsica d'aquesta llibreria és la de permetre realitzar gràfiques de molts tipus diferents en programes escrits en Java.

Aquesta està dissenyada per ser usada en aplicacions, applets [1], servlets [17] i JSP. JFreeChart no està orientada a l'usuari final, sinó que proporciona les eines necessàries als desenvolupadors per poder mostrar les gràfiques desitjades en les aplicacions.

2.2.1 Orígens i Requeriments

El projecte JFreeChart va ser fundat al Febrer de l'any 2000 per David Gilbert. Actualment l'utilitzen entre uns 40,000 i 50,000 programadors de tot el món i continua sent desenvolupat per David Gilbert amb contribucions de diverses comunitats de programadors de tot el món.

JFreeChart requereix de la plataforma Java 2 (JDK versió 1.3 o superior).

És important dir que JFreeChart es una llibreria per a programadors; no una aplicació per l'usuari final. El qual farà ús de les funcionalitats que el programa li proporcionen sent-li transparent l'utilització de la llibreria.

2.2.2 Característiques

JFreeChart pot generar gràfics circulars, gràfics de barres (regulars o acumulatius, amb efectes 3D opcionals), gràfics lineals, gràfics de dispersió, gràfics temporals, gràfics de punts, entre d'altres. Tots els quals estan degudament explicats a la documentació oficial de la llibreria.

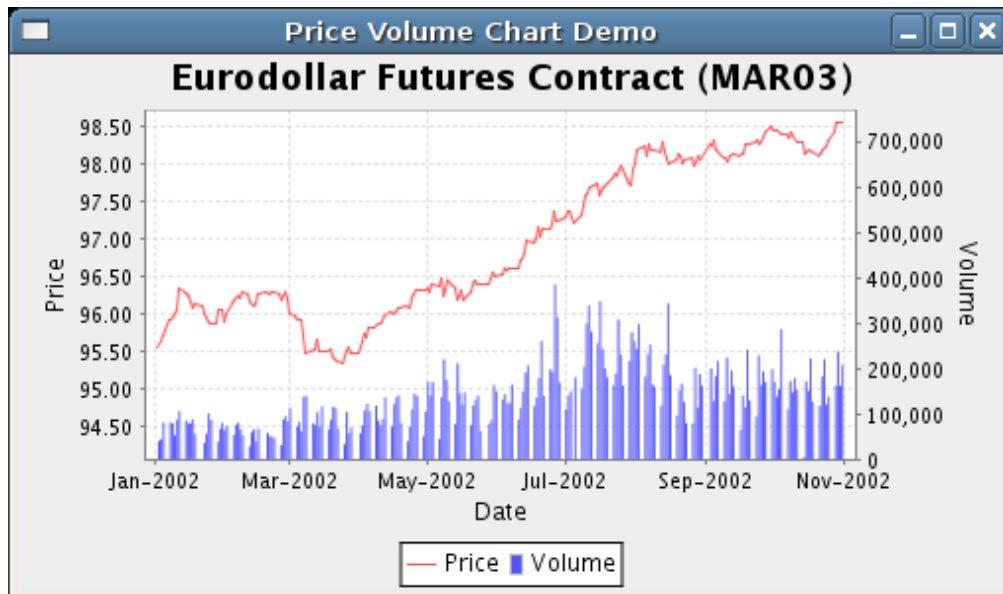


Figura 3: Exemple de gràfic generat amb JFreeChart

Com a recull de les principals característiques, podem enumerar:

- Les dades són accessibles des de qualsevol implementació de les interfícies definides.
- Permet exportar el gràfics als formats PNG, JPEG, PDF i SVG.
- Permet realitzar zoom interactiu sobre les gràfiques.
- Funciona en aplicacions, servlets, JSP i applets.
- El codi font es troba disponible sota la llicència GNU Lesser General Public License (LGPL).
- JFreeChart està escrit íntegrament en Java.

Cal comentar que JFreeChart es distribueix sota la llicència GNU Lesser General Public License (LGPL), malgrat això la guia oficial per desenvolupadors (JFreeChartDeveloperGuide), no és de lliure distribució sinó que és propietària i per tant de pagament.

A l'hora de treballar amb la llibreria, es té l'opció de comprar la documentació oficial o bé una altra opció perfectament vàlida és la d'obtenir multitud d'exemples que es troben a Internet o a llibres especialitzats amb l'objectiu d'entendre com funciona la llibreria.

La pàgina web oficial del projecte es pot trobar a:

<http://www.jfree.org/jfreechart/index.html>

En aquesta pàgina podem trobar informació recent sobre JFreeChart, incloent-hi exemples, enllaços per descarregar, Javadoc [9], fòrums de discussió i més documentació.

2.2.3 Exemple pràctic de JFreeChart sobre l'aplicació realitzada

En l'aplicació de simulació realitzada, ens va resultar necessari mostrar gràfics partint d'unes dades prèviament calculades. Tot seguit la figura 4 mostra un exemple extret de l'aplicació realitzada en el qual es mostra el codi necessari i el gràfic obtingut amb aquest codi.

```
for(Integer i=0;i<llistaResultats.size();i=i+2){

    dataset.setValue(Integer.parseInt((String) this.llistaResultats.get(i+1)),
    "Profit1", (String) this.llistaResultats.get(i) );

}

chart=ChartFactory.createBarChart3D(this.titol,this.titolX,this.titolY,dataset,PlotOrientation.VERTICAL,false,false,true);
```

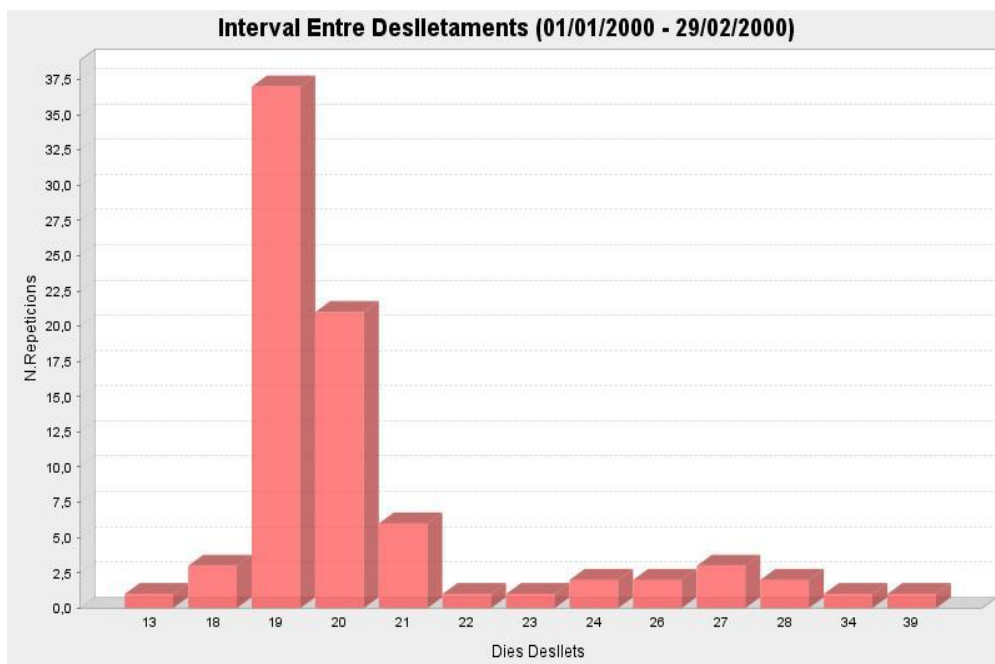


Figura 4: Gràfic de barres3D realitzat a l'aplicació de simulació

2.3 Java

Com tots sabem el Java és un llenguatge de programació desenvolupat per Sun Microsystems sobre l'any 1990 a partir de C++. Aquest llenguatge fou dissenyat per a ésser orientat a objectes; és a dir, que segueix la filosofia de programar mòduls senzills, per tal de crear aplicacions de més gran envergadura.

Un dels factors que ha portat a Java a ser un dels llenguatges més utilitzats actualment i de més prestigi, és que és un llenguatge potent però senzill i segur. Java aporta senzillesa ja que elimina moltes de les característiques d'altres llenguatges com c++, per mantenir reduïdes les especificacions del llenguatge i poder afegir característiques útils com el *garbage collector* [5] (reciclador de memòria dinàmica). El qual fa que no sigui necessari preocupar-se per alliberar la memòria, ja que el reciclador ja s'encarrega d'això i com és de baixa prioritat, quan entra en acció, permet alliberar blocs de memòria molt grans; la qual cosa limita en molt la fragmentació de la memòria.

També Java redueix en un 50% els errors més comuns en la programació com en llenguatges com ara el C, o el C++ ja que elimina moltes de les característiques d'aquest, entre les que destaquen:

- L'aritmètica de punters.
- No existeixen referències.
- Registres (struct).
- Definició de tipus (typedef).
- Macros (#define).
- Necessitat de alliberar memòria (free).

Un altre punt a tenir en compte és que Java és un llenguatge de codi obert sota la llicència GPL. Això junt amb l'afany de Sun i dels usuaris que han desenvolupat llibreries lliures per a Java, a fet que Java sigui un llenguatge de programació molt estès en l'actualitat.

Una altra de les peculiaritats de Java és que és un llenguatge interpretat i, per tant, pot semblar lent en comparació amb altres llenguatges, però ofereix un índex de reutilització del codi molt elevat. A més, és un llenguatge flexible i potent per la facilitat amb què es programa i els resultats que ofereix.

Un dels trets que el caracteritza i que fa que sigui una eina molt valorada a l'hora de desenvolupar aplicacions distribuïdes, és el fet que sigui un llenguatge multi plataforma.

Un altre detall interessant de Java és que simplifica enormement l'ús de protocols nous o actualitzats. Per exemple si executem una aplicació Java a la xarxa i troba una peça de l'aplicació que no sap utilitzar, Java és capaç de portar automàticament quelcom peça que el sistema necessiti per funcionar.

Observem les figures següents per tal de entendre la idea:

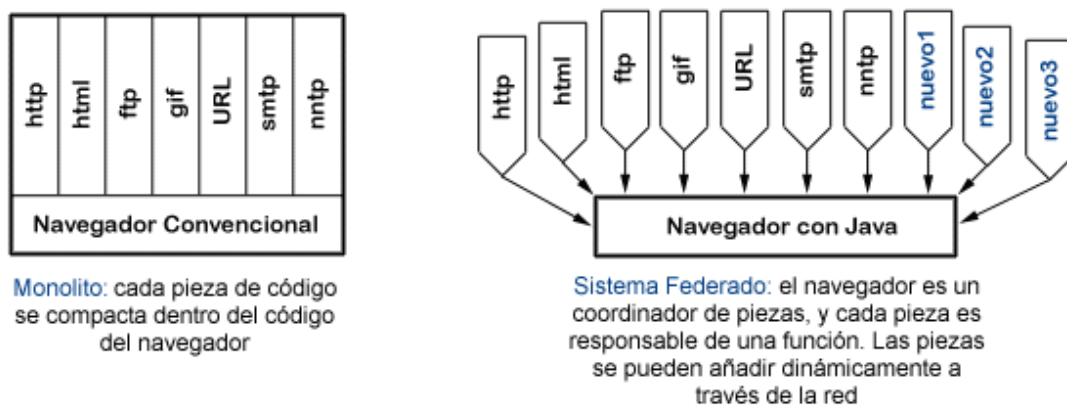


Figura 5: Dinamisme de Java en la Xarxa

Tot això fa que Java sigui un llenguatge molt dinàmic en la xarxa.

Sun Microsystems disposa de tres implementacions diferents de Java:

- J2SE per a aplicacions d'escriptori
- J2EE per a aplicacions distribuïdes (online)
- J2ME per a plataformes amb recursos més reduïts com ara mòbils o PDA.

Per a la nostra aplicació hem tractat la J2EE per l'aplicació Web i la J2SE per a l'aplicació d'escriptori. Tot i que en realitat, com ja explicarem més endavant, totes dues implementacions parteixen de la mateixa capa de domini [3]; i que en el nostre cas tan sols es modifica la capa interfície. Totes aquestes definicions sobre les capes i el disseny de l'aplicació ja s'explicarà més detingudament en capítols posteriors. Ja que per exemple en el cas de la interfície Web s'han utilitzat JSP (Java Server Pages) que és una tecnologia que permet desenvolupar pàgines web i generar respostes dinàmicament a peticions HTTP, permetent la incrustació de codi Java en un context estàtic (J2EE).

2.4 Apache Tomcat

Apache Tomcat, antigament anomenat sota el projecte Apache Jakarta; és un servidor web amb suport de servlets i JSP; es defineix també com un contenidor de servlets.

Un contenidor de Servlets, és un Shell d'execució que utilitza i invoca servlets per compte de l'usuari.

Podem dividir els contenidors de servlets en:

- Contenedors de servlets Stand-Alone (Independents)

Aquests són part integral del servidor web els quals utilitzen un servidor web basat en Java. Un exemple d'aquests contenidors és el contenidor de servlets que ahora és part de JavaWebServer (actualment substituït per iPlanet). Aquest és el mode per defecte que utilitza Tomcat.

- Contenedors de servlets dins de procés

El contenidor Servlet és una combinació de plugin per al servidor web i una implementació del contenidor Java. El plugin del servidor web obra una JVM (Màquina Virtual de Java) dins de l'espai d'adreces del servidor web i permet que el contenidor Java s'executi en ell. Si una certa petició s'hagués d'executar en un servlet, el plugin obtindria el control sobre la petició i ho passaria al contenidor Java (utilitzant JNI). Un contenidor d'aquest tipus és adequat per servidors multi fils d'un sol procés i proporciona un bon rendiment però està limitat en escalabilitat.

- Contenedors de servlets fora de procés

Com en el cas anterior el contenidor de servlet és una combinació de plugin per al servidor web i una implementació del contenidor Java; però en aquest cop s'executa una JVM fora del servidor web. El plugin del servidor web i el JVM del contenidor Java es comuniquen mitjançant IPC (normalment sockets TCP/IP). Si una certa petició hagués d'executar un servlet, el plugin obtindria el control sobre la petició i ho passaria al contenidor de Java (mitjançant IPCs). El temps de resposta en aquest tipus de contenidors no es tan bo com en l'anterior, però haurien d'obtenir-se millors rendiments en altres aspectes (escalabilitat, estabilitat, etc).

Tomcat pot utilitzar-se com un contenidor solitari (principalment pel desenvolupament i la depuració) o com a plugin per a un servidor web existent (actualment és suporta els servidors Apache, IIS i Netscape). Això significa que sempre que despleguem Tomcat haurem de decidir com utilitzar-lo, a més també necessitarem instal·lar un adaptador de servidor web (en les versions antigues que no s'inclouïa amb el paquet).

En els seus orígens Tomcat va començar sent una implementació de l'especificació dels servlets començada per James Duncan Davidson, que treballava com arquitecte de programari a Sun. És per aquest fet, que Tomcat implementa les especificacions de servlet i de Java Server Pages (JSP) de Sun Microsystems, proporcionant un entorn per al codi Java a executar en cooperació amb un servidor web.

Posteriorment va ajudar a fer el projecte OpenSource i en la seva donació a l' Apache Software Foundation.

Al començament d'aquest, va existir la percepció de que l'ús de Tomcat de forma autònoma era només recomanable per a entorns de desenvolupament, entorns amb requisits mínims de velocitat i per a gestió de transaccions. Avui dia ja no existeix aquesta percepció, i Tomcat és usat com servidor web autònom en entorns amb alt nivell de tràfic i alta disponibilitat.

Tomcat inclou el seu propi servidor HTTP, és per aquest fet que a l' inici l'hem definit com un servidor web. Pel que fa al motor de servlets de Tomcat sovint es presenta en combinació amb el servidor web Apache o altres servidors Web.

2.5 La Tecnologia Java Server Faces (JSF)

JavaServer Faces (JSF) és un framework per a aplicacions Java basades en web que simplifica el desenvolupament d'interfícies d'usuari del costat del servidor en aplicacions Java EE.

En la seva majoria, les aplicacions JavaServer Faces són com qualsevol altra aplicació Web implementada en Java. S'executen en un contenidor Servlet Java, i típicament contenen:

- Components JavaBeans [8] (objectes del model en tecnologia JavaServer Faces) contenint dades i funcionalitats específiques de l'aplicació.
- Escoltadors d'Esdeveniments.
- Pàgines, com a pàgines JSP.
- Classes d'utilitat del costat del servidor, com beans per a accedir a les bases de dades.

A més d'aquests ítems, una aplicació JavaServer Faces també té específicament:

- Una llibreria d'etiquetes personalitzades per a dibuixar components d'interfície d'usuari en una pàgina.
- Una llibreria d'etiquetes personalitzades per a representar manegadors d'esdeveniments, validadors, i altres accions.
- Components d'interfície d'usuari representats com objectes amb estat en el servidor.
- Validadors, manegadors d'esdeveniments i manegadors de navegació.

Aquest model de programació ben definit i la llibreria d'etiquetes per a components d'Interfície d'Usuari (d'ara en endavant UI) faciliten de forma significativa la tasca de la construcció i manteniment d'aplicacions Web amb UIs del costat del servidor.

Com es pot apreciar en la figura 6, la interfície [7] d'usuari que creem amb la tecnologia JavaServer Faces (representat per myUI en el gràfic) s'executa en el servidor i es renderitza en el client.

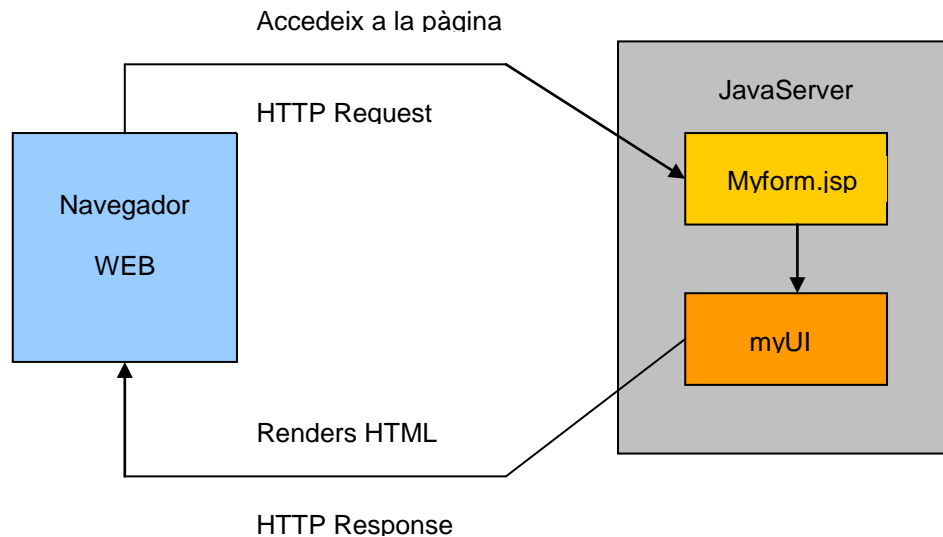


Figura 6: Model de funcionament de JSF

La pàgina JSP, myform.jsp, dibuixa els components de la interfície d'usuari amb etiquetes personalitzades definides per la tecnologia Java Server Faces.

El UI de l'aplicació Web (representat per myUI en la imatge) controla els objectes referenciats per la pàgina JSP:

- Els objectes components que mapegen les etiquetes sobre la pàgina JSP.
- Els receptors d'esdeveniments, validadors, i els conversors que estan registrats en els components.
- Els objectes del model que encapsulen les dades i les funcionalitats dels components específics de l'aplicació.

Així doncs una aplicació JavaServer Faces inclou:

- Un conjunt d'APIs per representar components d'una interfície d'usuari i administrar el seu estat, controlar esdeveniments, validar l'entrada, definir un esquema de navegació de les pàgines i donar suport per a internacionalització i accessibilitat.

- Un conjunt per defecte de components per a la interfície d'usuari.
- Dues llibreries d'etiquetes personalitzades per JavaServer Pages que permeten expressar una interfície Java Server Faces dins d'una pàgina JSP.
- Un model d'esdeveniments en el costat del servidor.
- Administració d'estats.
- Beans administrats.

En la figura 7 il·lustra el cicle de vida d'un JSF.

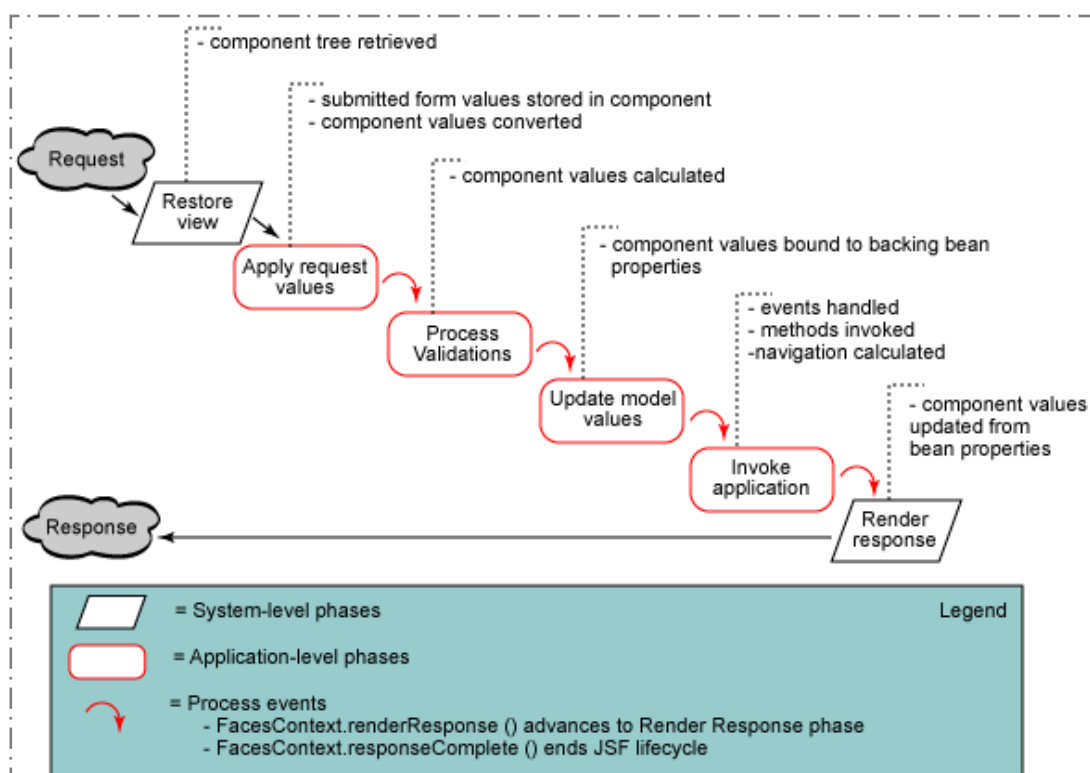


Figura 7: Cicle de vida d'un JSF

Amb una aplicació JSF resulta relativament fàcil:

- Connectar esdeveniments generats en el client amb el codi de l'aplicació del costat del servidor.
- Mapejar components UI a una pàgina de dades del costat del servidor.
- Construir un UI amb components reutilitzables i extensibles.
- Gravar i restaurar l'estat del UI més enllà de la vida de les peticions de servidor.

2.5.1 Model Vista Controlador (MVC)

Un dels patrons més coneguts en el desenvolupament web és el patró MVC (Model Vista Controlador). Aquest patró ens permet/obliga a separar la lògica de control (sap quines coses cal fer però no com), la lògica de negoci (sap com es fan les coses) i la lògica de presentació (sap com interactuar amb l'usuari). Utilitzant aquest tipus de patrons aconseguim: més qualitat, millor mantenibilitat, perdre la por al foli en blanc (es té un patró a seguir al començar un projecte), però una de les coses més importants és la normalització i estandardització del desenvolupament de Programari.

JSF (Java Server Faces) és un framework de desenvolupament basat en el patró MVC (Model Vista Controlador). JSF a més pretén normalitzar i estandarditzar el desenvolupament d'aplicacions web.

2.5.2 Beneficis de la Tecnologia JavaServer Faces

Una de les grans avantatges de la tecnologia JavaServer Faces és que ofereix una clara separació entre el comportament i la presentació. Les aplicacions Web construïdes amb tecnologia JSP aconseguen parcialment aquesta separació. No obstant això, una aplicació JSP no pot mapejar peticions HTTP controlant esdeveniments específics dels components o controlar elements UI com objectes amb estat en el servidor. La tecnologia JavaServer Faces ens permet construir aplicacions Web que implementen una separació entre el comportament i la presentació tradicionalment ofertades per l'arquitectura UI del costat del client.

La separació de la lògica de la presentació també li permet a cada membre de l'equip de desenvolupament d'una aplicació Web centrar-se en la seva part del procés de desenvolupament, i proporciona un senzill model de programació per a enllaçar totes les peces. Per exemple, els Autors de pàgines sense experiència en programació poden usar les etiquetes de components UI de la tecnologia JavaServer Faces per a enllaçar codi de l'aplicació des de dins de la pàgina Web sense escriure cap script.

Altre objectiu important de la tecnologia JavaServer Faces és millorar els conceptes familiars de component-UI i capa-Web sense limitar-nos a una tecnologia de script particular o un llenguatge de marques. Encara que la tecnologia JavaServer Faces inclou una llibreria d'etiquetes JSP personalitzades per a representar components en

una pàgina JSP, els APIs de la tecnologia JavaServer Faces s'han creat directament sobre el API JavaServlet. Això ens permet fer algunes coses com: fer servir una altra tecnologia de presentació al costat de JSP, crear els nostres propis components personalitzats directament des de les classes de components, i generar sortida per a diferents dispositius client.

Però el més important, la tecnologia JavaServer Faces proporciona una rica arquitectura per a manejar l'estat dels components, processar les dades, validar l'entrada de l'usuari, i manejar esdeveniments.

En resum podem dir que:

- JSF tracta la vista (l' interfície d'usuari) d'una forma una mica diferent al que estem acostumats en aplicacions web. Seria més similar a l'estil de Swing [18], Visual Basic o Delphi, on la programació de la interfície es fa a través de components i està basada en esdeveniments (es prem un botó, canvia el valor d'un camp, ...).
- JSF és molt flexible. Per exemple ens permet crear els nostres propis components, o crear els nostres propis “render” per a pintar els components segons ens convingui.
- És relativament senzill

2.5.3 Versions de JSF

JSF consta de 4 versions fins a la data, la versió utilitzada per l'aplicació de simulació realitzada en el TFC es la 2.0; ja que el també recent entorn de desenvolupament NetBeans IDE 6.0 ens ha proporcionat les eines per treballar-hi.

- JSF 1.0 – Versió final 2004-03-11.
- JSF 1.1 – Versió final 2004-05-27.
- JSF 1.2 – Versió final 2006-05-11.
- JSF 2.0 – Versió final 2008.

Capítol 3

Aplicació Realitzada

3.1 Arquitectura

Un dels aspectes importants a l'hora d'implementar un sistema de bases de dades és l'arquitectura del sistema en sí. Hi ha dues arquitectures bàsiques que utilitzen els SGBD (Sistemes de Gestió de Bases de Dades). L'arquitectura centralitzada i l'arquitectura distribuïda. Ara passarem a descriure breument les característiques de les dues, per a posteriorment centra-nos en la que s'ha implementat en la nostra aplicació.

3.1.1 BBDD (distribuïdes vs centralitzades)

Abans de res definirem que és un Sistema de Gestió de Bases de Dades o SGBD ja que està íntegrament relacionat amb les bases de dades.

Un SGBD (del Angles DataBase Management System), és una agrupació de software que serveix per definir, construir i manipular una base de dades.

Quan parlem de que el SGBD defineix una base de dades ens referim a especificar els tipus de dades, estructures i restriccions per les dades que s'emmagatzemaran a la BD. Al parlar de construir, ens estem referint al procés d'emmagatzemar les dades sobre algun mitjà d'emmagatzemament. I diem que manipula la base de dades, ja que inclou funcions per a realitzar consultes, actualitzacions, etc a la base de dades que gestiona. Dit això, ja podem endinsar-nos en les possibles arquitectures de les bases de dades.

Existeixen varies arquitectures, que passarem a descriure a continuació.

3.1.1.1 Les bases de dades Centralitzades (BDC)

Les BDC són les que tenen com a característiques que:

- Els usuaris accedeixen al sistema mitjançant terminals de computadores que no tenen capacitat de processament, només ofereixen capacitats de visualització.
- Tot el procediment es realitza de forma remota.
- Del sistema central als terminals de visualització només s'hi envia informació i controls de pantalla.
- Els terminals estan connectats al sistema central mitjançant diferents tipus de xarxes de comunicació.

A la figura següent es mostra l'estructura d'una base de dades centralitzada.

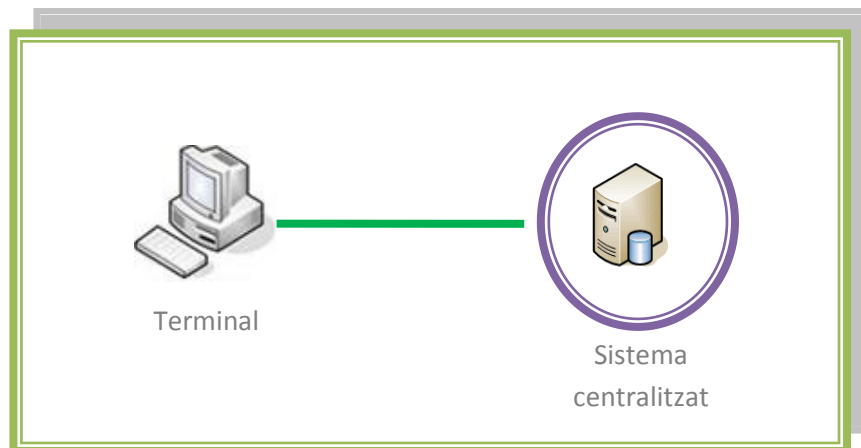


Figura 8: Sistema Centralitzat.

Cal comentar que el sistema centralitzat està format per dos grans pilars, el de Software i el de Hardware / Firmware [4]. L'esquema seria el següent:

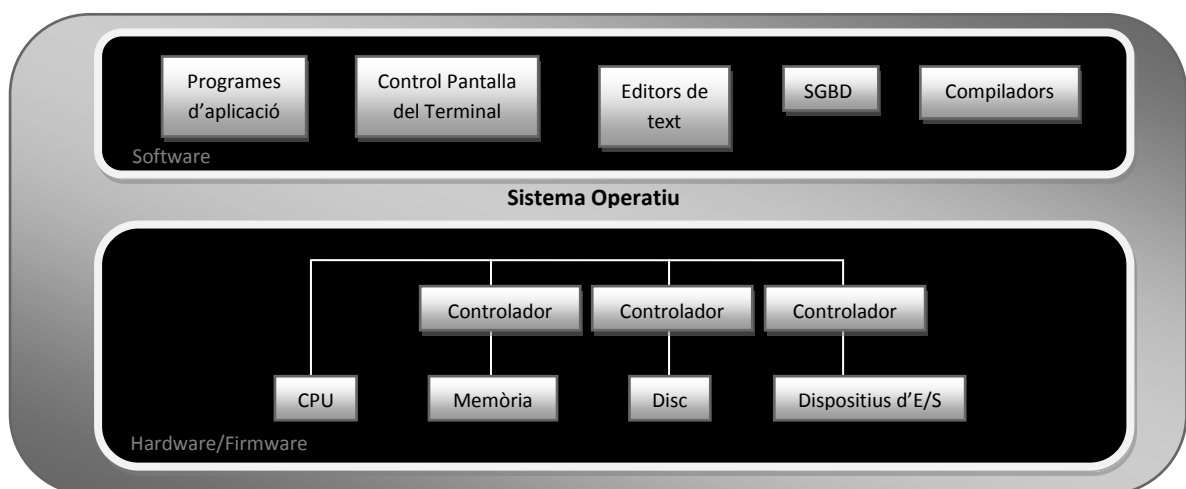


Figura 9: Sistema Centralitzat

3.1.1.2 Les bases de dades Distribuïdes (BDD)

Una BDD és una col·lecció de dades que lògicament pertanyen a un mateix sistema, però que físicament estan emmagatzemades en diferents instal·lacions d'una xarxa d'ordinadors.

Aquestes bases de dades, les BDD tenen dos aspectes molt importants a tenir en consideració:

- La distribució: El fet que les dades no s'emmagatzemin en la mateixa instal·lació, ens fa diferenciar les BDD de les BDC.
- La correlació lògica: El fet que les dades tinguin algunes propietats com les que s'emmagatzemen en una mateixa instal·lació ens fa diferenciar una BDD d'un conjunt de bases de dades (BD) locals que es troben emmagatzemades cada una en una instal·lació diferent d'una xarxa de computadors.

Un concepte bàsic sobre les bases de dades distribuïdes és que tan poden estar emmagatzemant les dades en un mateix lloc, com poden estar-ho fent en llocs geogràficament diferents.

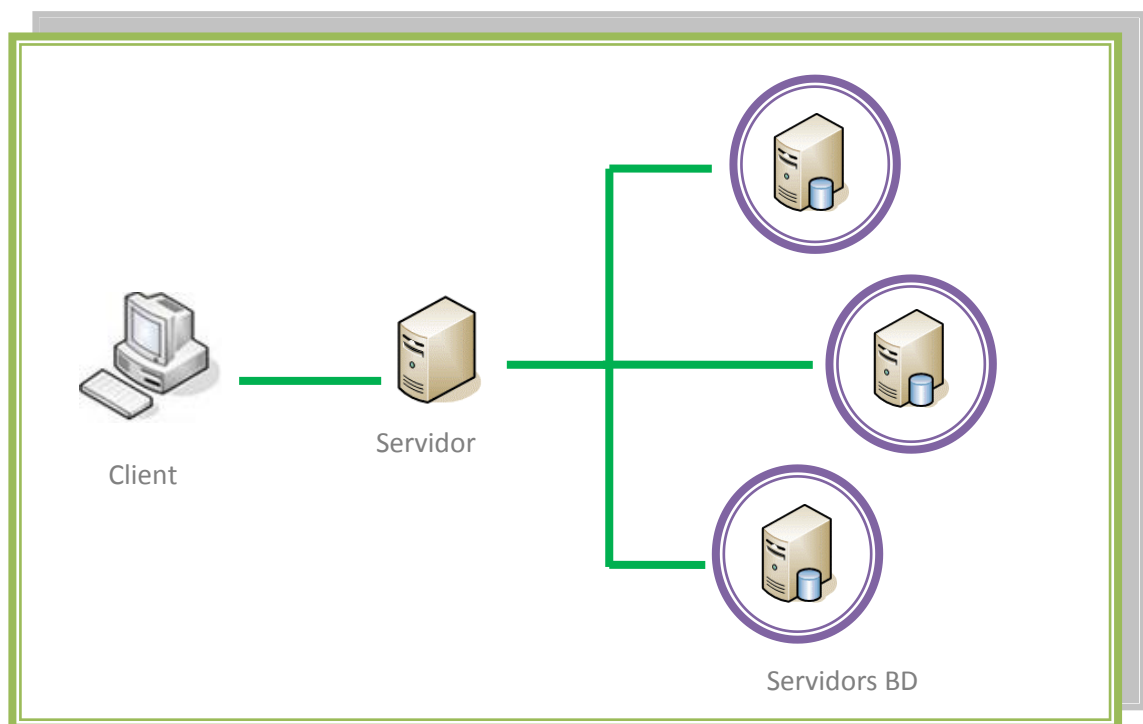


Figura 10: Arquitectura BDD amb servidors de BD en diferent localització.

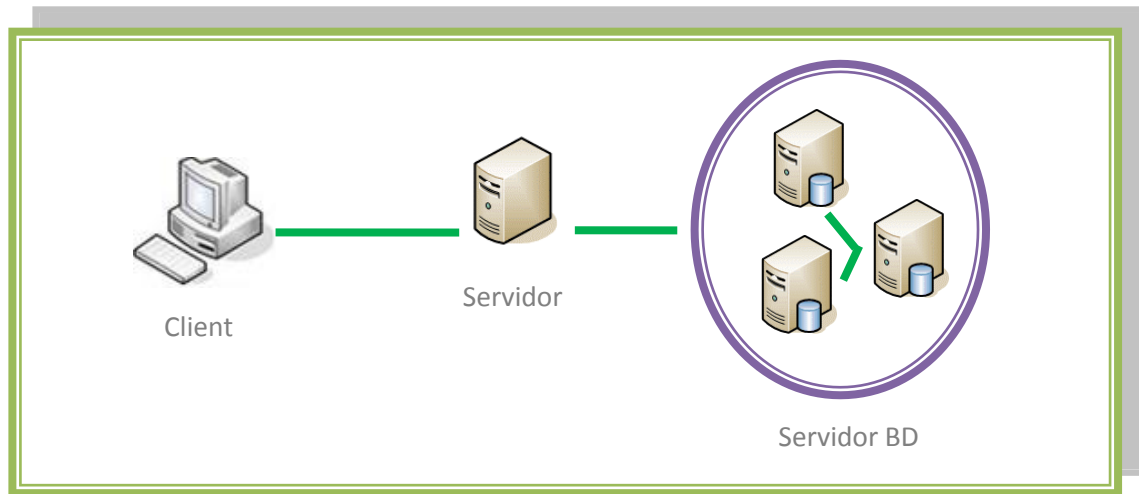


Figura 11: Arquitectura BDD amb servidors de BD en la mateixa localització.

En les dues figures anteriors s'ha mostrat un sistema de BDD de tres nivells on venen identificats per el client, el servidor d'aplicacions i els servidors de bases de dades. Cal tenir en compte que no forçosament ha d'esser així, podem dissenyar una BDD de tan sols dos nivells on existeixi un pel client i un altre pels servidors de bases de dades. També com s'ha pogut observar el fet de tractar amb arquitectures de bases de dades distribuïdes, ràpidament ho associem al disseny client – servidor. És en aquest, on es defineixen servidors especialitzats amb funcions específiques; els recursos que proporcionen els servidors especialitzats estan a disposició de un o molts clients; i els clients proporcionen a l'usuari les interfícies adequades per utilitzar els servidors, així com la potència necessària de processament per a tractar aplicacions locals.

Concretant definirem un sistema distribuït, com el conjunt d'instal·lacions amb capacitat de treballar interconnectades (sense perdre la capacitat de treballar de forma autònoma).

Definirem que una base de dades és centralitzada (BDC) quan totes les dades estan emmagatzemades en un mateix dispositiu.

I finalment definirem una base de dades distribuïda (BDD) quan la base de dades està físicament distribuïda entre diferents llocs d'un sistema distribuït.

3.1.2 Model Client/Servidor

Aquesta arquitectura consisteix bàsicament en que un programa (el client), realitza peticions a un altre programa (el servidor) que li dóna resposta. Encara que aquesta idea es pot aplicar a programes que s'executen sobre una sola computadora és més avantatjosa en un sistema operatiu multi usuari distribuït a través d'una xarxa d'ordenadors.

Partint de la idea del Model Client/Servidor i aplicant-ho a l'aplicació de simulació realitzada en aquest TFC, s'observa com l'idea de Client/Servidor està força present. L'aplicació realitzada es basa en dues eines software que són la Base de Dades MySQL i el propi programa escrit en llenguatge Java desplegat amb el servidor de Servlets/JSP Apache Tomcat, en el cas de l'aplicació web.

Com que en aquest projecte s'ha realitzat l'aplicació de simulació, tant amb una interfície web com amb una interfície no web basada en Java Swing, podem trobar diferències en el concepte de model client/Servidor de cadascuna.

Primer ens centrarem en conèixer com és l'arquitectura client/servidor de l'aplicació amb interfície basada en Java Swing (no web). Coneixent l'estructura de l'aplicació, observem que la Base de Dades MySQL seria el servidor de Dades de l'aplicació i el client seria l'aplicació Java amb interfície Swing.

La figura 12 mostra esquemàticament el funcionament de les peticions client/servidor de l'aplicació d'escriptori basada en Java Swing.

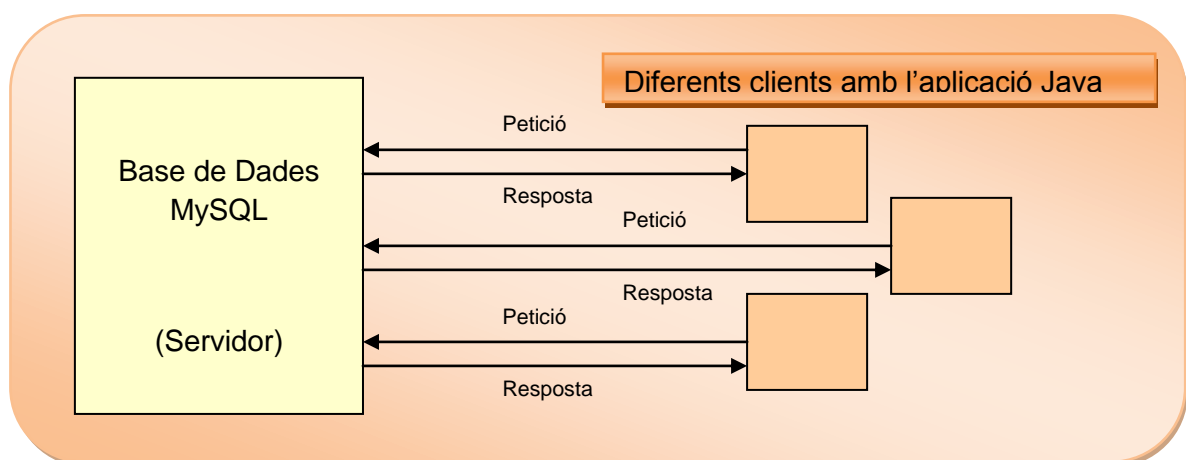


Figura 12: Peticions client/servidor aplicació no web

Ara ens centrarem en conèixer com és l'arquitectura client/servidor de l'aplicació amb una interfície Web. Al igual que en l'aplicació no web, la Base de Dades MySQL és el servidor de Dades de l'aplicació, però ara ens apareix també un altre servidor que alhora és client de la base de dades, és a dir el servidor de Servlets/JSP Apache Tomcat. El qual agafa el rol de client de la base de dades, ja que realitza consultes sobre ella i alhora fa de servidor sobre els navegadors web que li fan peticions per desplegar l'aplicació web.

La figura 13 mostra el funcionament de les peticions client/servidor de l'aplicació web i ens permet diferenciar el funcionament amb l'aplicació d'escriptori.

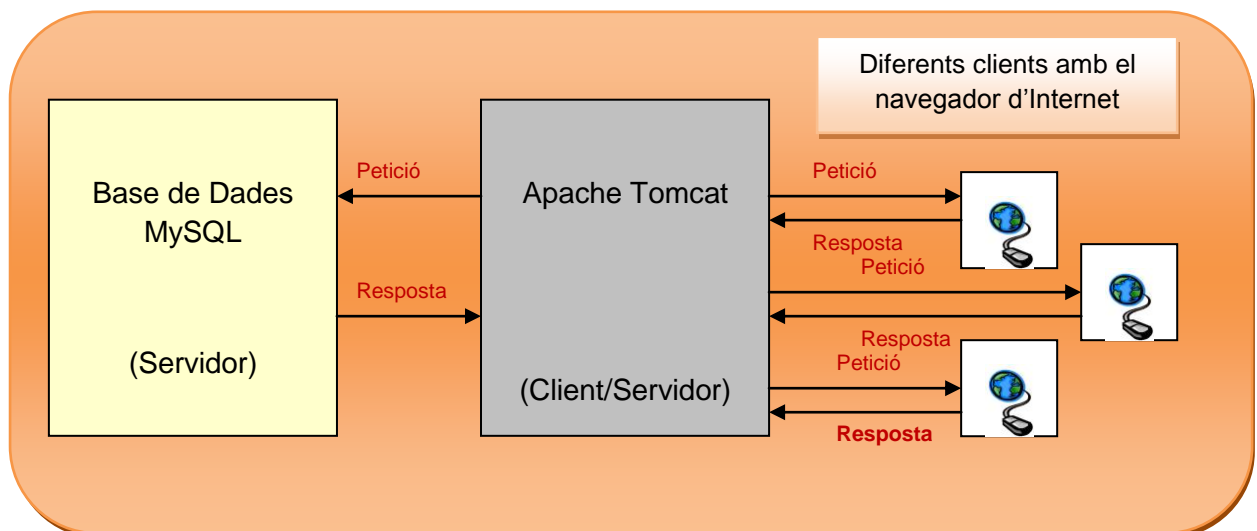


Figura 13: Peticions client/servidor aplicació web

3.1.2.1 Avantatges i versatilitat del model Client/Servidor

En aquesta arquitectura, la capacitat de procés està repartida entre els clients i els servidors; encara que són més importants els avantatges de tipus organitzatiu deguts a la centralització de la gestió de l'informació i la separació de responsabilitats, el que facilita i clarifica el disseny del sistema.

La separació entre client i servidor és una separació de tipus lògic, on el servidor no s'executa necessàriament sobre una sola màquina ni és necessàriament un sol programa.

Una disposició molt comuna són els sistemes multicapa en els quals el servidor es descompon en diferents programes que poden ser executats per diferents computadores, augmentant així el grau de distribució del sistema.

Podem enumerar dos punts clau de les avantatges de l'arquitectura client/servidor:

- **Centralització del control:** els accessos, recursos i la integritat de les dades són controlats pel servidor de manera que un programa client defectuós o no autoritzat no pugui danyar el sistema.
- **Escalabilitat:** es pot augmentar la capacitat de clients i servidors per separat.

Cal tenir en compte la gran versatilitat que ens ofereix el model client/servidor, el qual ens permet decidir on volem executar els diferents servidors.

En el cas del programa web realitzat, una bona decisió seria muntar el servidor de base de dades en un computador amb una capacitat de càlcul elevada ja que les consultes SQL són computacionalment costoses, el servidor Apache Tomcat requerirà probablement menys potència així que podríem decidir dedicar un computador més modest a fer aquesta tasca i finalment els clients que es connectarien per fer servir l'aplicació, els quals simplement es dedicarien a enviar peticions i rebre informació.

3.1.3 Multi plataforma

L'aplicació realitzada ha estat desenvolupada bàsicament com ja hem vist en apartats anteriors basant-nos en una base de dades MySQL i el llenguatge de programació Java, fent servir Apache Tomcat per desplegar l'aplicació web.

En aquest apartat el que es vol mostrar és que totes les eines fetes servir permeten que l'aplicació resultant sigui totalment multi plataforma, és a dir podem fer-la servir a pràcticament qualsevol Sistema Operatiu.

Si ens referim a la tecnologia Java, sabem que és un llenguatge interpretat el qual s'executa sobre una màquina virtual de Java; per tant a qualsevol entorn sobre el que treballem i disposem d'una màquina virtual Java, el programa serà executable.

El Servidor Apache Tomcat disposa de versions pre-compilades tant per Linux com per Windows així com del codi font disponible, per tant també estem parlant d'un programa multi plataforma.

Finalment fixant-nos en base de dades MySQL, tenim disponibles també versions per a múltiples Sistemes Operatius entre ells els més comuns com són: Linux, Windows i MacOS.

Tenint en compte el descrit anteriorment, podem assegurar que l'aplicació realitzada és totalment multi plataforma. Aquesta característica és de gran utilitat ja que ens permet per exemple en l'aplicació de simulació realitzada, instal·lar el servidor de Base de Dades en un entorn Linux, l'Apache Tomcat en un entorn Windows i accedir mitjançant un navegador Web des de qualsevol Sistema Operatiu.

La figura 14 mostra l'exemple descrit de com podríem distribuir el desplegament de l'aplicació en diferents Sistemes Operatius.

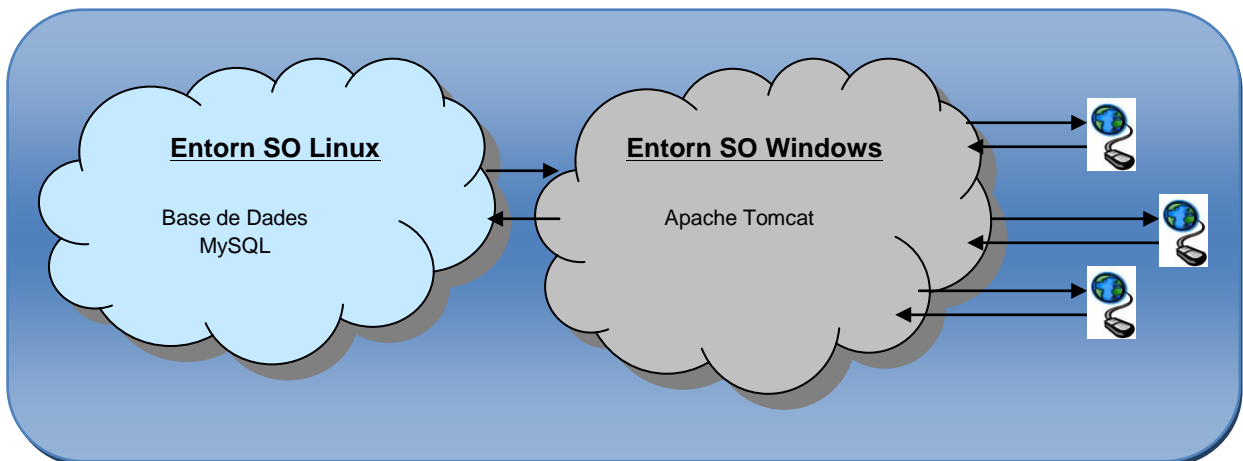


Figura 14: Exemple interacció entre diferents S.O.

És important valorar aquest aspecte ja que no sempre es pot triar l'entorn sobre el què es vol treballar o es volen instal·lar els servidors i el fet de poder tenir prou flexibilitat com per adaptar-se és força interessant.

3.2 Capes

La programació per capes és un estil de programació en la que el objectiu primordial és la separació de la lògica de negocis, de la lògica de disseny; un exemple bàsic d'això és el separar la capa de dades, de la capa de presentació a l'usuari.

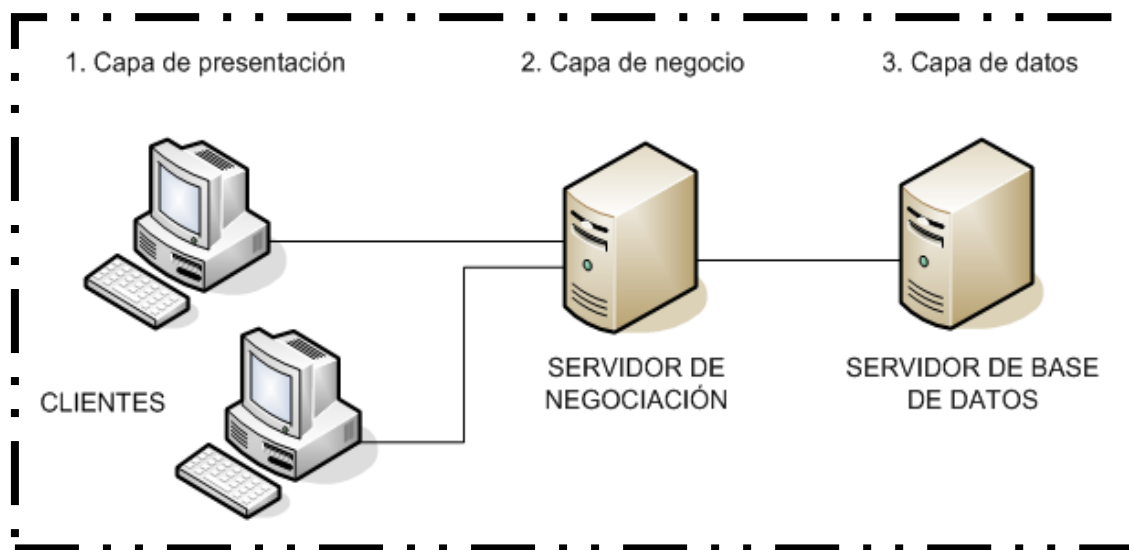


Figura 15: Representació de la programació per capes (capa presentació, domini i dades).

L'avantatge principal d'aquest estil de programació, és que el desenvolupament es pot portar a terme en diferents nivells i en cas de que es realitzi algun canvi en l'aplicació, tan sols involucra a la capa requerida i no cal revisar tot el codi.

A més, permet distribuir la tasca de la creació de l'aplicació per nivells; d'aquesta manera, poden existir diferents grups de treball i aquests estar totalment abstractes de la resta de nivells; essent simplement necessari conèixer la API (Application Programming Interface) que existeix entre nivells.

A més de poder treballar amb diferents grups de desenvolupament, una altra de les avantatges de la programació en capes, és la de que permet que els dissenys de les nostres aplicacions siguin més escalables. Això farà que es puguin ampliar amb facilitat en cas de que les necessitats de l'usuari augmentin. Tot això sense necessitat

de haver de dissenyar un altre cop el disseny i començar de zero, ni de resseguir tot el codi per aplicar una modificació.

Un bon disseny en capes, ens suposarà un cost inicial major; però a llarg termini esdevindrà a ser la millor opció.

En l'actualitat el disseny en tres nivells (o en tres capes) es el més utilitzat en les aplicacions tant d'escriptori com web.

3.2.1 Domini

Com tot disseny d'una bona aplicació és molt important dissenyar bé les capes i concretar exactament que hi ha d'haver implementat en cadascuna d'elles.

Seguin les referències que dicten la bona implementació, hem de separar el codi en dos grans blocs (el tercer bloc es el de la capa de dades i d'això ja s'encarrega la bases de dades MySQL que ja hem definit prèviament).

Els blocs han de ser diferents i ben diferenciats entre sí. Estem parlant de la capa domini i la capa presentació. Aquestes capes a grans trets ens vindrien a separar el que és codi de l'aplicació amb el que és l'interfície.

El domini s'encarregaria de la posta en funcionament i l'execució de totes les funcionalitats de l'aplicació, gestionant també els possibles errors que pugui cometre l'usuari final; i la capa presentació s'encarregaria únicament de com, on i de quina manera es mostra la informació que ha estat resultat dels càlculs de la capa domini.

Cal tenir en compte que és molt important no barrejar la capa domini, amb la capa presentació. La raó es obvia, el fet de treballar en capes és per a que si en un futur es vol canviar el medi de visualització de l'aplicació. Si tota la funcionalitat es troba en la capa domini, tan sols ens haurem de preocupar de implementar una nova capa interfície utilitzant la funcionalitat ja implementada.

A continuació podem observar un esquema bàsic del disseny de programació en capes.

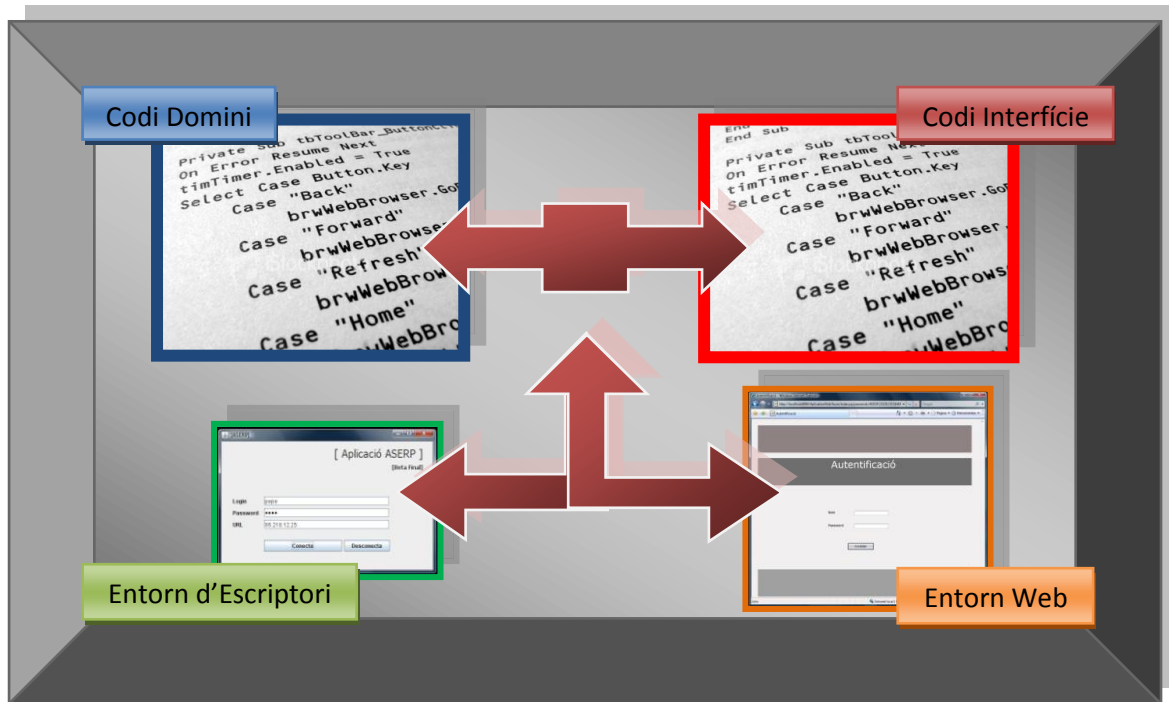


Figura 16: Esquema funcionament aplicació per capes

Una cosa que s'ha de tenir en compte per a no provocar la confusió, es no confondre el terme "nivell" amb capa.

La capa, correspon a l'idea de segmentar la solució (el seu codi) des d'un punt de vista lògic:

- Capa presentació.
- Cama domini.
- Capa dades.

Mentre que el terme nivell, correspon a la forma en que les capes lògiques es troben distribuïdes de forma física.

Per exemple:

- Una solució de tres capes (presentació, domini i dades) que resideixen en un sol ordinador (presentació+domini+ dades), es diu que la seva arquitectura és de tres capes en un sol nivell.
- Una solució de tres capes (presentació, domini i dades) que resideixen en dos ordinadors (presentació+domini, domini+ dades), es diu que la seva arquitectura és de tres capes en dos nivells.
- Una solució de tres capes (presentació, domini i dades) que resideixen en tres ordinadors (presentació, domini, dades), es diu que la seva arquitectura és de tres capes en tres nivells.

3.2.2 Interfície

Com ja s'ha dit anteriorment, l'aplicació realitzada està formada per dues capes, la capa Domini i la capa interfície, en aquest apartat desenvoluparem l'explicació de les interfícies, tant en la seva forma web (JSF) com en la seva forma d'aplicació en Java(Swing).

En el context del procés d'interacció persona-ordinador, la interfície gràfica d'usuari, és l'artefacte tecnològic d'un sistema interactiu que possibilita, a través de l'ús i la representació del llenguatge visual, una interacció amigable amb un sistema informàtic.

La interfície gràfica d'usuari (en anglès Graphical User Interface, GUI) és un tipus d'interfície d'usuari que utilitza un conjunt d'imatges i objectes gràfics per a representar la informació i accions disponibles en la interfície. Habitualment les accions es realitzen mitjançant la manipulació directa per a facilitar la interacció de l'usuari amb l'ordinador.

Una particularitat del projecte realitzat és que amb la mateixa capa domini s'han realitzat dues interfícies completament diferenciades, la primera que s'explicarà a continuació serà la interfície no web basada en la llibreria Java Swing i tot seguit s'explicarà la interfície web basada en JSF.

3.2.2.1 Interfície Swing

Swing és una biblioteca gràfica per Java que forma part de les Java Foundation Classes (JFC). Inclou components per a la interfície gràfica d'usuari tals com caixes de text, botons, desplegable i taules.

Característiques de les interfícies basades en Swing

Swing és una plataforma independent per Java. Segueix un model de programació per fils, i conté les següents característiques:

- **Independència de plataforma:** Swing és una plataforma independent de la plataforma sobre la que es vol executar.
- **Extensibilitat:** Swing és una arquitectura altament particionada que permet la utilització de diferents plugins; Els usuaris poden realitzar les seves pròpies implementacions modificades per a sobreesciure les implementacions per defecte. En general, els usuaris de swing poden estendre el framework per a: estendre classes existents i proveir alternatives d'implementació per a elements essencials.
- **Orientat a components:** Swing és un framework basat en components. La diferència entre objectes i components és un punt bastant subtil: concisament, un component és un objecte de bona conducta amb un patró conegut i especificat característic del comportament.
- **No utilització de components nadius del S.O.:** Per a la representació dels components Swing utilitza les apis 2D de Java i no directament les apis que proporcionen els diferents S.O.

Com totes les decisions d'implementació que es prenen a l'hora de programar, l'ús de la llibreria Swing proporciona una sèrie d'avantatges i una sèrie d'inconvenients.

Avantatges:

- El disseny en Java pur posseeix menys limitacions de plataforma.
- El desenvolupament de components Swing es més actiu.
- Els component de Swing suporten més característiques que altres llibreries gràfiques.

Desavantatges:

- La majoria dels navegadors no inclouen classes swing, pel que és necessari utilitzar un plugin Java. L' utilització d'un plugin i que es pugui mostrar al navegador no vol dir que l'aplicació es pugui considerar una aplicació web, per això en aquest projecte s'han realitzat les dues interfícies ja esmentades on la interfície JSF realitzada, que s'explicarà en el següent punt si permet parlar d'aplicació web.
- Els components swing generalment són més lents i presenten més problemes degut al fet que estan fets en Java pur, i solen presentar problemes relacionats amb vídeo en diverses plataformes.
- No sempre tenen el mateix aspecte que en el sistema on van ser dissenyats.

Metodologia de treball i exemples

Swing es basa en components, i de components n'hi ha de molts tipus; per exemple un botó de l'aplicació seria un component, un quadre de text seria un altre component entre molts altres que proporciona la llibreria.

Cada component té una sèrie d' esdeveniments ja definits que es criden quan l'esdeveniment d'un component es accionat, per exemple un component botó tindrà un esdeveniment que serà l'acció de pitjar aquest botó (depenent del tipus de component podem tenir diferents esdeveniments), de forma que quan l'esdeveniment s'activi tindrà un codi associat que s'executarà com a resposta a l' esdeveniment.

Podem dir que la programació d'interfícies en Swing es una programació orientada a esdeveniments, quan es genera un esdeveniment, s'executa el codi controlador d'aquest. Els components swing poden tenir associats un o més esdeveniments, depenent de les especificacions de cada component.

La llibreria Swing es troba àmpliament documentada en la seva documentació oficial, la intenció d'aquest apartat no és la de mostrar tots els components i els esdeveniments associats amb aquests que proporciona la llibreria swing ja que són bastants i no es pretén fer de manual, sinó explicar en que consisteix i com funciona per comprendre millor el projecte, si es vol aprofundir més també es pot mirar el codi de la interfície de l'aplicació realitzada el qual també és proporciona amb aquest projecte.

Tot seguit veurem un exemple de l'explicació sobre components i esdeveniments realitzada.

La figura 17, mostra un exemple de com a l'aplicació realitzada, el botó etiquetat com a Connecta i anomenat ConfirmaUsuari al ser activat per l'usuari llança un esdeveniment i executa un codi associat amb aquest esdeveniment:

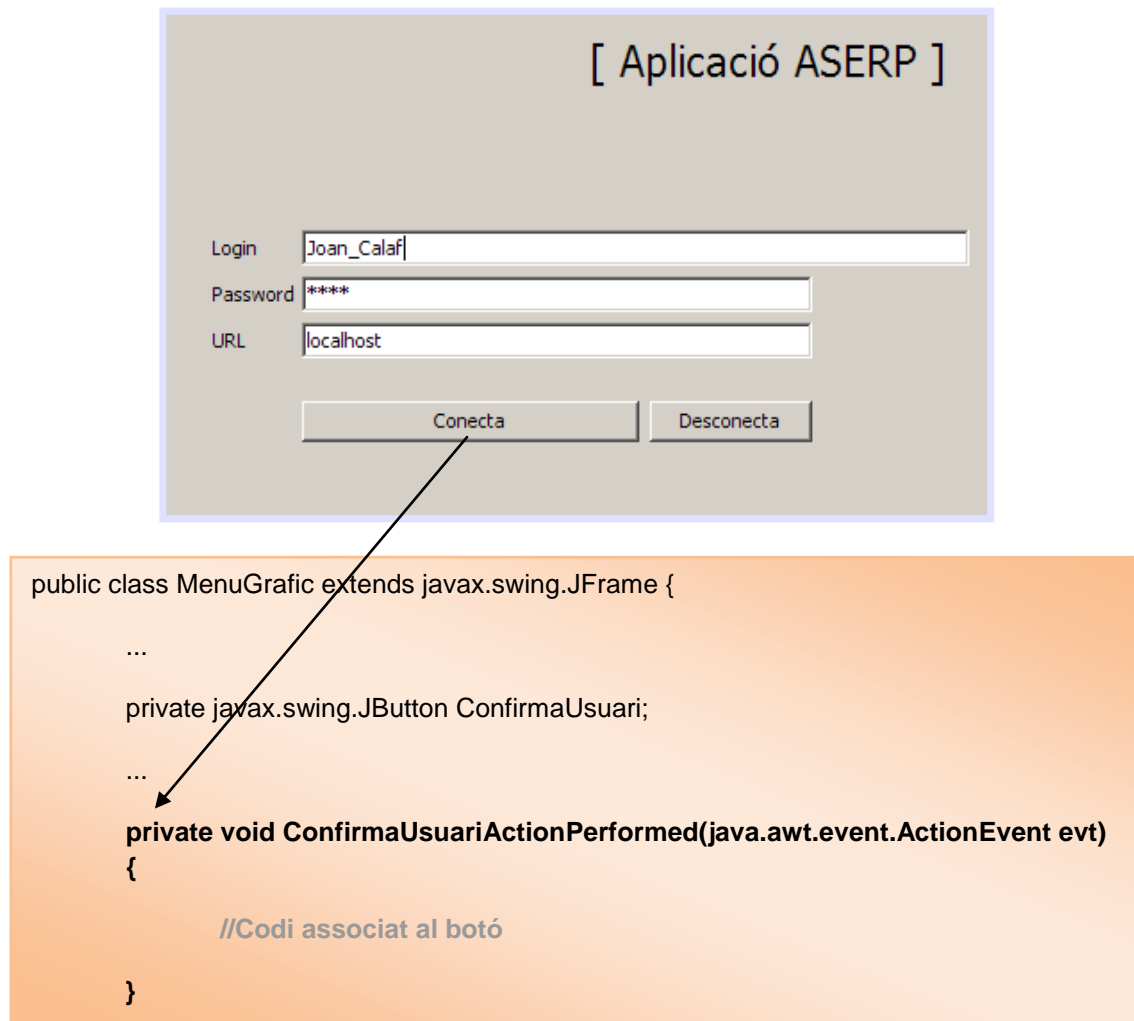


Figura 17: Exemple botó Swing

3.2.2.2 Interfície JSF

L'explicació tècnica sobre Tecnologia Java Server Faces (JSF) ja ha estat realitzada en l'apartat 2.5, el que es veurà a continuació en aquest apartat és perquè s'ha triat la tecnologia JSF i com ha estat realitzada la nostra aplicació web.

La necessitat d'una interfície Web i la tria de JSF

Un dels objectius de l'aplicació de simulació realitzada des del començament era que aquesta aplicació fos una aplicació web. La versió amb swing realitzada no ens permetia dir que aquesta aplicació fos web i tampoc ens aportava la flexibilitat d'una aplicació web, ja que aquesta no podia ser executada des de qualsevol navegador i requeria tenir instal·lat el J2SE Java Runtime Environment (JRE).

Com que ja teníem realitzada la interfície de l'aplicació en Swing, varem provar de crear una interfície basada en JavaScrip Pages(JSP). El que ens varem trobar va ser que ens resultava molt costos e incòmode de treballar amb JSP ja que no ens proporcionava els components necessaris per poder portar l'aplicació Swing al seu equivalent amb JSP. El treball ens resultava molt lent pel fet de no disposar d'un entorn de desenvolupament que ens proporcionés una interfície gràfica per treballar amb els components gràfics. Vist això varem decidir realitzar la interfície web amb JSF ja que el concepte de JSF és més similar al de Swing tal i com està explicat amb més detall al capítol 2 apartat 5 , JSF ens proporcionava components de tipus Botó, quadre de text, calendaris desplegable, etc..., cadascun dels quals tenia definits uns esdeveniments d'una forma molt similar a swing i es per això que varem triar JSF.

Un altre aspecte a destacar es que l'entorn de desenvolupament Netbeans IDE 6.0 que ja estàvem utilitzant per a la implementació de la funcionalitat de l'aplicació, proporciona suport nadiu a JSF de forma que realitzar el desenvolupament és molt més amigable i estalvia el fet d'haver de programar tot en mode text.

Metodologia de treball i exemples

A l'hora de realitzar l'interfície amb JSF, bàsicament teníem tres elements a controlar per a poder fer funcionar correctament l'aplicació, els elements són els següents:

- **Fitxer JSP:** Encarregat de definir els atributs de la pàgina web com els color de fonts etc..., i també de definir els components (no la funcionalitat) i la seva posició a la pàgina. Cada nova pagina que es crea te el seu corresponent fitxer JSP.
- **Fitxer Java:** Encarregat de contenir la implementació dels esdeveniment dels components definits al fitxer JSP. Cada nova pagina que es crea te el seu corresponent fitxer Java.
- **Fitxer XML(faces-config.xml):** Encarregat de definir l'esquema de navegació entre les diferents pàgines de la interfície web. Es únic per a tota la interfície.

Tot seguit les figures 18, 19 i 20 mostren un exemple del codi que conté cada fitxer per al final mostrar i dotar de funcionalitat un botó.

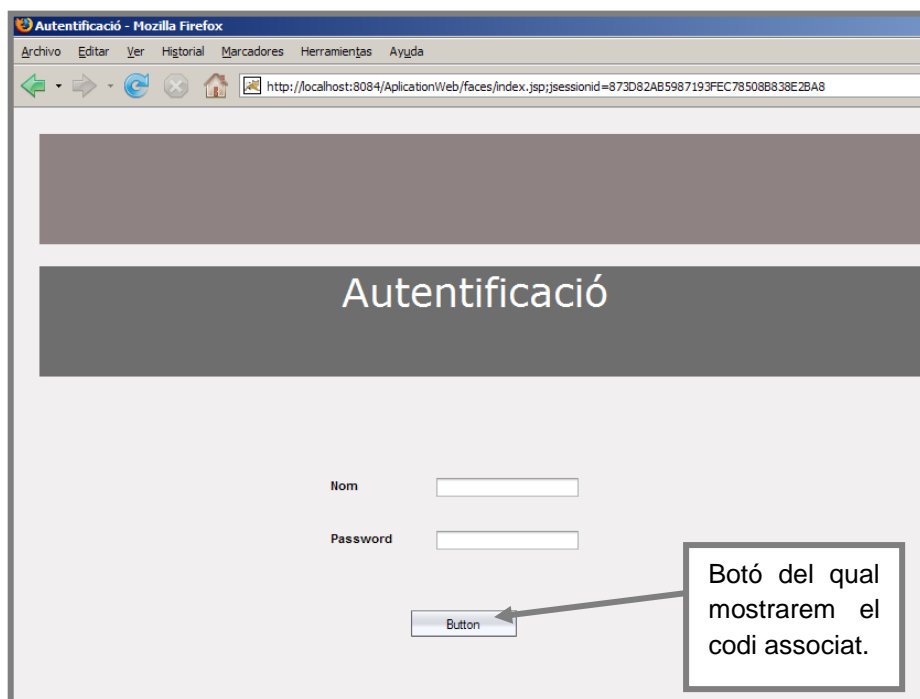


Figura 18: Exemple botó JSF

Aquest quadre mostra en negreta el codi del fitxer JSP corresponent al component botó que es veu a la figura 18.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Document   : AutenticacioBase
  Created on : 19-ene-2008, 10:46:06
  Author    : luis
-->
<div style="background-color: rgb(241, 239, 239); height: 200px; width: 400px; -rave-layout: grid"
xmlns:f="http://java.sun.com/jsf/core" xmlns:webuijsf="http://www.sun.com/webui/webuijsf">
  <f:subview id="AutenticacioBase">
    <webuijsf:textField binding="#{AutenticacioBase.nomf}" id="nomf" style="height: 24px; left: 168px;
top: 48px; position: absolute; width: 120px"/>
    <webuijsf:passwordField binding="#{AutenticacioBase.passwf}" id="passwf" style="height: 24px; left:
168px; top: 96px; position: absolute; width: 120px"/>
    <webuijsf:button actionExpression="#{AutenticacioBase.acceptarbuton_action}"
binding="#{AutenticacioBase.acceptarbuton}" id="acceptarbuton"
    style="position: absolute; left: 144px; top: 168px; width: 96px; height: 24px" text="Button"/>
    <webuijsf:staticText binding="#{AutenticacioBase.nom}" id="nom" style="font-weight: bolder; height:
24px; left: 72px; top: 48px; position: absolute; width: 72px"/>
    <webuijsf:staticText binding="#{AutenticacioBase.password}" id="password" style="font-
weight: bolder; height: 24px; left: 72px; top: 96px; position: absolute; width: 72px"/>
  </f:subview>
</div>
```

Figura 19: Codi del fitxer JSP

Aquest quadre mostra en negreta el codi del fitxer Java corresponent al component botó que es veu a la figura 18.

```
public class AutenticacioBase extends AbstractFragmentBean {

  public String aceptarbuton_action() {

    // Codi que es vulgui executar al pitjar el botó

    return "case1";

  }
```

Figura 20: Codi del fitxer Java

Finalment el fitxer faces-config.xml contendria la informació corresponent a la navegació entre pàgines, en aquest cas la informació del fitxer ens diria que si **acceptarbuton_action()** ha retornat **"case1"**, enllaçaríem amb la següent pàgina web que en l'aplicació realitzada seria la de funcionalitats.

Cal comentar que amb l'entorn de desenvolupament Netbeans 6.0 que hem utilitzat, se'ns ha facilitat força el treball ja que proporciona una interfície de disseny gràfic on un cop el programador ha dissenyat l'aplicació (quins components conté i on van col·locats), sol ha de programar el codi dels components, el fitxer JSP es genera automàticament i sol cal modificar-lo a mà per fer coses molt puntuals.

3.3 L10n

En els dies que corren en que la tecnologia i la informació està a l'abast de tothom seria d'ingenus no aprofitar l'oportunitat de mostrar els productes a tot el món.

Per tal de que la nostra aplicació sigui accessible a més gent hem realitzat l'aplicació web que compleixi l'anomena't l10n.

Degut a que els nostres propòsits dels projecte no eren la realització d'una web, sinó els de fer l'aplicació per a la simulació; no s'ha rematat la feina fent que la web complís els criteris d'internacionalització o també anomenats i18n. Tot i així, hem cavat uns bons fonaments per tal de que algú altre pugui continuar el nostre projecte.

Diferències entre Localització i Internacionalització

Cada persona té les seves pròpies definicions preferides per aquests termes. Nosaltres oferirem aquí algunes descripcions generals, d'alt nivell, que indiquen el mode en que acostumen a utilitzar aquests termes en el lloc del W3C dedicat a la internacionalització.

Localització

S'entén per localització la adaptació d'un producte, aplicació o contingut d'un document amb la finalitat d'adequar-ho a les necessitats (lingüístiques, culturals, entre d'altres) d'un mercat destinatari concret.

La paraula localització a vegades s'escriu com "l10n", on 10 es la quantitat de lletres entre la "l" i la "n".

Encara se la considera a vegades sinònim de traducció de l' interfície d'usuari i de documentació, la localització acostuma a ser un assumpte considerablement més complex i que pot implicar l'adaptació del contingut amb:

- Formats numèrics, de data i hora.
- Us de símbols de la moneda.
- Us del teclat.
- Algoritmes de comparació i ordenament.
- Símbols, icones i colors.
- Texts i gràfics que continguin referències a objectes, accions o idees que, en una cultura determinada, puguin ésser objecte de mala interpretació o considerats ofensius.
- Diferents exigències legals.
- Entre d'altres.

La localització pot requerir inclús una reelaboració exhaustiva de la lògica del disseny visual o la presentació, si la forma de fer negocis (per exemple, les normes contables) o el paradigma acceptat d'aprenentatge (per exemple, èmfasi en l'individu o en el grup) en la localitat de destí, si són molt diferents en relació amb la cultura originària.

Internacionalització

Existeixen diferents definicions per a la paraula internacionalització. Algunes persones utilitzen altres termes per a referir-se al mateix concepte, per exemple, "globalització". La internacionalització és el disseny i desenvolupament d'un producte, una aplicació o el contingut d'un document de manera tal que permeti una fàcil localització amb destinació a audiències de diferents cultures, regions o idiomes. La paraula internacionalització de vegades s'escriu "i18n".

La internacionalització generalment implica:

- Una manera de disseny i desenvolupament que elimini obstacles a la localització o la distribució internacional. Això inclou qüestions tals com (entre altres) usar *Unicode* o assegurar, allí on correspongui, un correcte tractament de les codificacions de caràcters antiquats; controlar la concatenació de cadenes; o evitar que la programació depengui de valors de cadenes pertanyents a la interfície d'usuari.

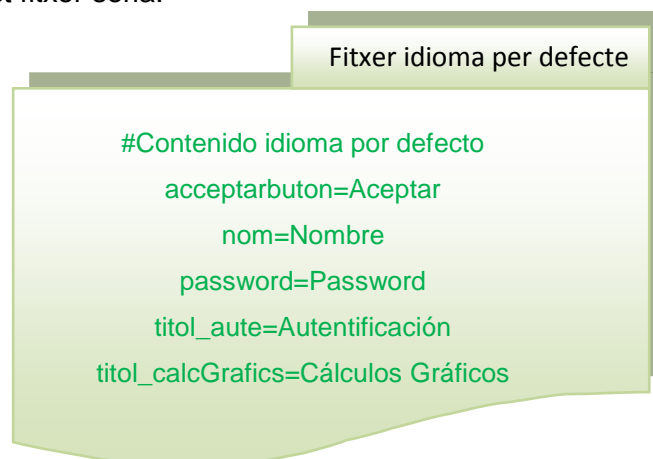
- Habilitar característiques que tal vegada no siguin usades fins al moment de la localització. Per exemple, afegir en la DTD etiquetes per a habilitar el text bidireccional o la identificació d'idiomes. O fer la CSS compatible amb text vertical o altres característiques tipogràfiques alienes a l'alfabet llatí.
- Preparar el codi per a plantar cara a les preferències locals, regionals, lingüístiques o culturals. En general, això suposa incorporar característiques i dades de localització predefinits a partir de biblioteques existents o de les preferències de l'usuari. Alguns exemples són: formats de data i hora, calendaris locals, formats i sistemes de nombres, ordenament i presentació de llistes, ús de noms personals i formes de tractament, etc.
- Separar del codi o contingut font els elements localitzables, de manera que puguin carregar-se o seleccionar alternatives localitzades segons determinin les preferències internacionals de l'usuari.

Com ja hem dit, la nostra aplicació no compleix al 100% tots els requisits del i18n, però sí els de l10n. La nostra aportació s'ha realitzat de la següent manera:

Primer de tot s'han etiquetat cadascun dels text (estàtics, labels, botons, etc) un cop ja havent identificat cadascun dels elements que volem que s'adaptin a la localització, hem creat els fitxers properties amb les parelles de nom ("*<nom_etiqueta>=<valor_etiqueta>*").

Un cop creats, el primer que s'ha de decidir és quin serà l' idioma per defecte de la nostra aplicació. El nostre ha estat per defecte el castellà.

Un exemple d'aquest fitxer seria:



D'aquest fragment del fitxer, comentar que en les properties quelcom cosa que escriguem a continuació del símbol coixinet “#” s'interpretarà com a comentari.

S'haurà de realitzar un fitxer per cadascun dels idiomes que vulguem que tingui la nostra aplicació. En el nostre cas hem adaptat la versió en català, castellà i anglès.

Els fitxers properties tenen una peculiaritat, i es que no es pot assignar el seu nom de quelcom manera.

Si per exemple al fitxer per defecte l'anomenem “MessagesBundle.properties” la resta s'hauran de dir “MessagesBundle_xx_YY.properties” on les “xx” faran referència a l'idioma i les “YY” al país concret on es parla.

Nosaltres hem realitzat els següents fitxers:

MessagesBundle.properties	(Fitxer per defecte).
MessagesBundle_es_ES.properties	(Fitxer idioma castellà).
MessagesBundle_es_CA.properties	(Fitxer idioma català).
MessagesBundle_en_GB.properties	(Fitxer idioma anglès).

I per a que la nostra aplicació els sàpigues interpretar hem cridat a la classe Java ResourceBundle per a que accedís als fitxers i ens retornes el valor de la etiqueta desitjada, amb el idioma que hem seleccionat a l' inici de la aplicació.

```

idioma=this.getSessionBean1().getId idioma();
if(idioma.equals("Spanish")){
    bundle = ResourceBundle.getBundle("MessagesBundle",new java.util.Locale("es","ES"));

}else if(idioma.equals("Catalan")){
    bundle = ResourceBundle.getBundle("MessagesBundle",new java.util.Locale("es","CA"));
}else if(idioma.equals("English")){
    bundle = ResourceBundle.getBundle("MessagesBundle", Locale.UK);
}
this.calcular.setText(bundle.getString("Calcular"));
this.desvTipica.setText(bundle.getString("DesvTipica"));
this.varianca.setText(bundle.getString("Varianca"));
this.mitjana.setText(bundle.getString("Mitjana"));
this.valor.setText(bundle.getString("Valor"));
this.valor2.setText(bundle.getString("Valor"));
this.ndeslletats.setText(bundle.getString("Ndeslletats"));

```

Figura 21: Fragment de codi del fitxer CalculsBase.java

Com s'ha pogut observar l' idioma es passat a través d'un bean de sessió que ha emmagatzemat el identificador del idioma seleccionat.

Codi involucrat en pitjar els botons d'idiomes:

```
public String catalan_action() {  
    this.getSessionBean1().setIdioma("Catalan");  
    return "case1";  
}  
  
public String spanish_action() {  
    this.getSessionBean1().setIdioma("Spanish");  
    return "case1";  
}  
  
public String english_action() {  
    this.getSessionBean1().setIdioma("English");  
    return "case1";  
}
```

Figura 22: Fragment de codi del fitxer index.java

Una altra opció hagués estat enlloc d'utilitzar els fitxers properties implementar classes d'idiomes (amb el mateix que hem dit pels nom dels fitxers) i que entenguessin de la classe ListResourceBundle.

A continuació es mostra un exemple del que també es podia haver realitzat.

```
import java.util.*;

public class Etiquetas_es_ES extends ListResourceBundle
{
    public Object[][] getContents()
    {
        return contenido;
    }
    private Object[][] contenido = {
        { "etiqueta_boton_abrir", "Abrir"},
        { "etiqueta_boton_cerrar", "Cerrar"},
        { "etiqueta_error", "Error en la aplicacion"}
    };
}
```

Figura 23: Alternativa als fitxers properties.

3.4 Diagrames

3.4.1 Diagrama Cas d'ús

3.4.2 Diagrama de Classes

3.5 Descripció i Format del fitxer VSI

Format lineal de la taula Altas:

0	5	11	13	14	16	22	27	33	34	40	42	128
Granja	Animal	Cicle	Sexe	Raça	Fecn	Origen	Fece	Baixa	Fecb	Raó	Blancs	

GRANJA: N° de Granja o Identificador de Granja. Es un camp de 5 dígits que ens indica el identificador de granja.

ANIMAL: N° de Animal. Es un camp de 6 dígits, distribuïts en tres subcamps:

- **Any:** Última xifra de l'any de la data de naixement (1 dígit)
- **Ordre:** En el cas de que un numero d'animal es repetís el número d'ordre seria el que marcaria la diferencia. Per defecte es posa aquest camp a 0. (1 dígit)
- **N° de l'animal :** Número d'identificació del reproductor. (4 dígits)

CICLE: N° de cicle. Ens indica el número de cicle en el que es troba l'animal en el moment de l'alta. Normalment es 00. (2 dígits)

SEXE: Sexe de l'Animal. Es un número identificador del sexe de l'animal. (1 dígit)

- 1 = Femella.
- 2 = Mascle.

RAÇA: Raça de l'Animal. Camp de 2 dígits numèrics que indica la raça del **pare** i la **mare** del reproductor en aquest mateix ordre.

- 1 = Large White.
- 2 = Landrace "matern".
- 3 = Landrace Alemà.
- 4 = Landrace Belga.
- 5 = Duroc.
- 6 = Pietrain.
- 7 = Hampshire.
- 8 = (altra raça).
- 9 = animal creuat.

FECN: *Data de Naixement*. Camp de 6 díigits numèrics per a la data del naixement.

ORIGEN: *Origen*. Camp de 5 díigits alfanumèrics per a indicar la granja d'origen de l'animal.

FECE: *Data d'Entrada en l'Explotació*. 6 díigits numèrics per a la data d'entrada en l'explotació (DDMMAA).

BAIXA: *Baixa*. Camp d'una posició numèrica que s'activa o no depenent si l'animal està o no donat de baixa.

- 0 = Actiu.
- 1 = Baixa.

FECB: *Data de Baixa*. Camp de 6 díigits numèrics que ens indica la data de la baixa.

RAÓ : *Raó de la Baixa*. Camp numèric de 2 posicions numèriques que indica el motiu de la baixa, segons sigui el codi:

Femelles		Mascles	
01	No embarassada després de les cubricions	31	Mort
02	No apareix en cel	32	Infertilitat
03	Avortada	33	Subfertilitat (sota del % de fertilitat)
04	Baixa productivitat per l'Edat	34	Dificultat en l'acoblament
05	Problemes de mal de peus	35	Camada petita
06	Morta en l'explotació	36	Aploms
07	Baix nombre de nascuts	37	Coixesa o artritis
08	Baix nombre de deslletats	38	Produeix nats morts
09	Accidents al part	39	Vell i pesat
10	Trastorns en la lactació	40	Consanguinitat (tares, ...)
11	Malaltia Respiratòria	41	Dolent
12	Infermetat declaració Obligatòria		Altres causes

13	Evaginació del Úter o del Recte		
14	Anomalies en els garrins		
15	Consanguinitat de la camada		
16	Garrins nascuts amb poc pes		
17	Males canals en descendents		
18	Males condicions de mercat		
19	Venda com reproductors		
20	Altres causes		

Taula 1: Raó de la baixa del fitxer VSI

Format lineal de la taula Altas:

0	5	11	13	14	20	26	32	38	44	50	56	62	68	74	75	76	77	78	79	85	86	88	90	91	93	95	96	101	103	128
---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----	-----	-----

GRANJA: *Nº de Granja.* Camp 5 dígits alfanumèrics que representa la granja en que es troba l'animal.

ANIMAL: *Número de l'animal.* Camp de 6 dígits alfanumèrics, distribuïts en tres subcamps:

- **Any:** Última xifra de l'any de la data de naixement (1 dígit)
- **Ordre:** En el cas de que un numero d'animal es repetís el número d'ordre seria el que marcaria la diferencia. Per defecte es posa aquest camp a 0. (1 dígit)
- **Nº de l'animal :** Número d'identificació del reproductor. (4 dígits)

CICLE: *Número de cicle.* Es el número de cicle en que s'està entrant. Camp de 2 dígits numèrics.

REP: *Número de repeticions.* Són les repeticions que s'han realitzat en un mateix cicle. (1 dígit)

DATA CUB.: *Data de Cubrició.* Hi ha 5 camps numèrics para la data de cubrició de 6 dígits, amb el que hi ha un màxim de 4 repeticions de 6 dígits numèrics cadascuna que representen la data de la cubrició. (DDMMAA).

MASCLE: *Número de Mascle.* Hi ha 5 camps alfanumèrics per a l'identificació del mascle al qui correspon la data de cubrició. Cada camp té 6 dígit alfanumèrics

TIPUS DE CUBR: *Tipus de Cubrició.* Camp de 1 dígit numèric que recull el codi del tipus de la cubrició, segons la llista següent:

- 1 = simple, sol una munta.
- 2 = múltiples montes amb un sol mascle.
- 6 = inseminació artificial exterior.
- 8 = múltiples montes amb mascles diferents.
- 9 = inseminació artificial.

DATA PART: *Data de Part.* Camp de 6 dígit numèrics que contenen la data del part. (DDMMAA).

TIPUS DE PART

0	---> Part Normal.
1	---> Avortament.

VIU: *Garrins Nascuts Vius.* Camp de 2 dígit numèrics que indiquen el número de garrins (mascles i femelles) nascuts vius.

MOR: *Garrins Nascuts Morts.* Camp numèric de 2 dígit que indiquen el número de garrins nascuts morts.

MOMI: *Garrins Nascuts Momificats.* Camp numèric d'un dígit que indica el número de momificats.

ADO: *Garrins Adoptats.* Camp numèric de 2 dígit que indica el nombre de garrins adoptats.

RET: *Garrins Retirats.* Camp numèric de 2 dígit que indica el nombre de garrins retirats.

DATA DESLLETAMENT: *Data del Deslletament.* Camp de 6 dígit numèrics per a la data de deslletament.

Nº GARRINS: *Nombre de Garrins Deslletats.* Camp numèric de 2 dígit que indica el nombre de garrins deslletats.

3.6 Implementació

Pel que fa a la implementació del software implementat (Aplicació Alta/Baixa usuaris del servidor, i l'aplicació ASERP tant en mode web com en mode escriptori) passarem a comentar els mètodes desenvolupats per tal de definir els dissenys elegits i el perquè d'aquests.

3.6.1 Aplicació Alta/Baixa d'usuaris del servidor de BD

Aquesta aplicació ha estat implementada per automatitzar i facilitar la feina del administrador de la base de dades en la gestió dels usuaris d'aquesta. Els dos mètodes principals són els de alta i baixa d'usuari, tot i que cadascun d'aquests es poden desgranar en submètodes per tal de dinamitzar el funcionament de l'aplicació. D'aquesta manera intentàvem que fos el codi més entenedor i al ser més granulat resultaria més senzill d'introduir noves funcionalitat a aquest.

En el cas de l'alta se l'hi ha d'introduir el nom de l'usuari que volem crear, el seu password i l'identificador de la granja que te assignada.

Aquest mètode internament realitza un conjunt de submètodes que serien els que es descriuen a continuació.

Primerament s'establiria la connexió a la base de dades.

Seguidament per tal de no poder donar d'alta dos usuaris amb el mateix nom, s'executaria una comprovació que consultaria a la BD que l'usuari que es vol donar d'alta sigui únic i no existeixi cap altre amb el mateix nom. Si sigues el cas que ja existís un altre amb el mateix nom, el sistema ens indicaria que ja es troba registrat un usuari amb el mateix nom. Acte seguit anul·laria el procés de l'alta.

Si hagués estat el cas en que l'usuari fos únic i no existís cap prèviament amb el mateix nom, continuaria amb el procés d'alta i crearia la base de dades juntament amb les taules que l'hi corresponen per a poder interactua amb l'aplicació ASERP en quelcom de les seves versions.

Un cop s'ha creat tot l'anterior es crearia l'usuari a la BD, se l'hi assignarien els permisos per poder interactuar amb la BD creada per ell, i s'introduiria l'usuari en una taula de control, juntament amb el seu password i l'identificador de la granja.

Si tot s'ha realitzat correctament, ja arribaria el torn de finalitzar la connexió i es tancaria el procés de l'alta.

```
AltaUsuari(nom,pass,granja)
    EstableixConexio()
    ComprobaRegistre(nomUser)
    CreateDataBase(nomUser)
        CreaTaules()
    CreateUser(nomUser,passUser,granja)
        IntroUserTaula(nomUser,passUser,granja)
    FinalitzaConexio()
```

Figura 24: Procés d'alta d'un usuari al servidor de BD.

El cas de baixa vindria a ser com l'inversa del procés d'alta. Inicialment se l'hi passaria el nom de l'usuari que es vol donar de baixa a l'aplicació i l'ApliServer establiria una connexió a la BD com a súper usuari, comprovaria que l'usuari existís (Si no existís trauria un missatge d'error confirmant-ho i anul·laria l'operació de baixa) i primerament eliminaria la base de dades d'aquest per a després eliminar l'usuari del sistema i de la taula de registres. Si tots els passos s'han realitzat correctament l'ApliServer realitzaria una desconexió de la BD per a finalitzar tot el procés.

```
BaixaUsuari(nom)
    EstableixConexio();
    ComprobaRegistre(nomUser)
    DeleteDataBase(nomUser);
    DeleteUser(nomUser);
        DropUserTaula(nomUser)
    FinalitzaConexio();
```

Figura 25: Procés d'alta d'un usuari al servidor de BD.

3.6.2 Aplicació ASERP (Paquet Domini)

En aquest cas igual que en l'anterior ens limitarem a definir el disseny del paquet domini de l'aplicació ASERP ja que és comú tan en la versió Web com en la versió d'escriptori.

Tota la implementació de la funcionalitat de l'aplicació es troba al paquet domini, i aquest està compost pels següents fitxers en l'estructura que es defineix a continuació:

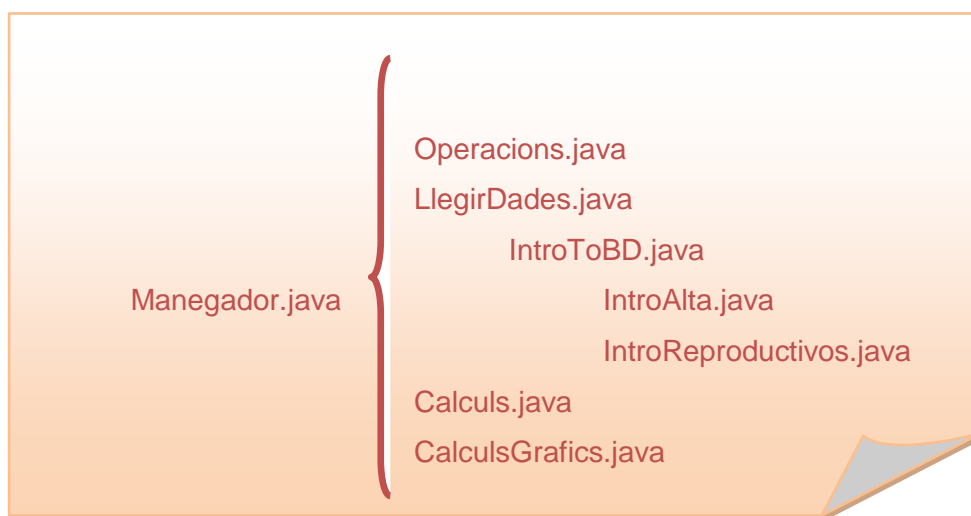


Figura 26: Arquitectura de la funcionalitat de l'ASERP.

Ara passarem a definir la tasca que realitza cadascun de les classes resultants després de la compilació dels fitxers mostrats en la figura anterior.

El **manegador** per definir-ho d'algun mode, és l'enllaç entre la capa de domini i la capa presentació. Vindria a ser la classe a partir de la qual es pot accedir a tota la funcionalitat de l'aplicació.

La classe **Operacions** al nostre parer és una de les classes amb més ingeni que hem implementat. Ja que a partir d'una sèrie de mètodes molt senzills és capaç de realitzar quelcom consulta a una base de dades indiferentment de la dificultat d'aquesta i de la longitud de filtratge de el camp "where" (recordem que la nostra BD és relacional i funciona sota SQL) i d'una manera totalment genèrica. A més a més, ens permet recuperar els resultats de la consulta en diferents estructures de dades (ResultSet o List).

La classe **LlegirDades** és l' inici del procés de lectura del fitxer VSI (fitxer que conté la informació de totes les granges que es volen introduir a la base de dades pel seu posterior estudi). Aquesta classe va capturant línia a línia el fitxer VSI i el transmet a la classe **IntroToBD** que aquesta a la vegada, després de comprovar el seu format per identificar si es informació destinada a la taula "altas" o "reproductivos" el passarà a la classe **IntroAlta** o **IntroReproductivos** depenen correspongui a un format o altre de dades.

Les classes **IntroAlta** i **IntroReproductivos**, finalment rebran la línia corresponent a la seva taula i en el format que tenen especificat i aquestes seran les encarregades de introduir cadascun dels valors a la BD. Per fer-ho aniran capturant variable a variable que es troben en cadascuna de les línies i fent les seves comprovacions corresponents per emmagatzemar-ho a la BD en el format que li toca; han de poder ser guardades com a variables de tipus text, de tipus enter o de tipus data. Aquest últim tipus si el fitxer, no continguís cap valor per la variable indicada el sistema introduiria una data per defecte que és la següent 01/01/1970. El perquè és molt simple, MySQL no ens permet tenir un camp de tipus data buit en cap de les taules.

La classe **Càlculs** proporciona un conjunt de càlculs ja predefinits, que han estat indicats per el corresponent estudi de simulació de les granges.

Alguns dels exemples d'aquests càlculs implementats (fent les corresponents consultes a la BD a partir d'una instància a un objecte de la classe operacions) són:

ProdTrujaPresent, ProdTrujaProductiva, GarrinDesII, Vius, Morts, Retiraments, Deslletaments, Avortaments, GarriDesIIPerTruja, IntervalEntreParts, EdatDeslletament, NombrePrimeresGarrinades, EdatPrimerCobriments, EdatPrimerPart, EdatTrujesPart, NombreTrujesBaixa, EdatTrujesBaixa, NombreTrujes, NombreBerrosPresents, TrujesPerBerro.

La classe **CalculsGrafics** al igual que l'anterior, es tracta d'un conjunt de càlculs ja predefinits però aquest cop gràfics.

Estan classificats per quatre tipus diferents de gràfics.

1. Els de **Interval** (Gràfics de mitjanes, que corresponen a l'estudi per duració d'interval. Exemple: Interval de deslletaments).
2. Els per **dia** (Gràfics amb valors molt concrets, valor exacte dia a dia Ex: Garrins Vius en una setmana concreta).
3. Els per **setmana** (Gràfics amb valors concrets, i al igual que l'anterior amb valors exactes però mesurant en setmanes. Ex: Estudi dels Garrins Vius en setmanes).
4. Els per **cicle** (Gràfic que no es centra en una estructura de data concreta, analitza tot el període introduït per igual, fent èmfasi en els diferents cicles de vida dels animals de la granja).

Alguns dels exemples d'aquests càlculs gràfics serien els següents:

ViusSetmanaGrafic, MortsSetmanaGrafic, RetiramentsSetmanaGrafic, NumPartsSetmanaGrafic, NCubSetmanaGrafic, AdoptatsSetmanaGrafic, NDeslletamentsSetmanaGrafic, NDeslletamentsDiaGrafic, NPartsDiaGrafic, NViusDiaGrafic, NMortsDiaGrafic, NViusCicleGrafic, NDeslletatsCicleGrafic, NAnimalsCicleGrafic, IntervalDeslletamentGrafic i IntervallIDCFGrafic

Aquests proporcionen una idea més intuïtiva de la simulació de la granja, ja que es poden comparar càlculs gràfics iguals o diferents per un mateix o distint període de temps. D'aquesta manera la comprensió del rendiment productiu de cadascuna de les granges serà més senzill. Sobretot pels usuaris que disposin de diferents granges pertanyents a una mateixa persona (usuari).

I ja per finalitzar comentar que existeixen diferents tipus d'usuaris, els que tenen més privilegis poden consultar dades ja siguin numèriques o gràfiques d'una o més (fins i tot de totes) granges que han estat introduïdes a la BD. Els usuaris més limitats tan sols podran fer estudis de les granges a les quals se'ls hi hagi donat permís per treballar.

3.7 Funcionalitats

En aquest apartat, s'explicaran pas a pas les funcionalitats que proporcionen les aplicacions creades per tal de fer funcionar l'aplicació de simulació ASERP realitzada.

Primerament s'explicarà el funcionament de l'aplicació encarregada de les altes i les baixes dels usuaris, seguidament es procedirà a explicar les funcionalitats compartides de l'aplicació web i de l'aplicació d'escriptori, les quals són pràcticament totes, ja que estem parlant de la mateixa aplicació amb diferents interfícies. Finalment es comentaran aquells aspectes que pel fet de tractar-se d'una aplicació web o per tractar-se d'una aplicació d'escriptori són exclusius d'aquestes.

3.7.1 Aplicació de gestió d'usuaris

La funcionalitat d'aquesta aplicació és la de donar altes o baixes amb la finalitat de que els usuaris puguin entrar al sistema ASERP.

Al obrir l'aplicació de gestió d'usuaris, el primer que se'ns demanarà és l'usuari i password d'administrador d'usuaris de l'aplicació ASERP, ja que cal tenir en compte que sol l'administrador d'usuaris podrà donar altes i baixes als usuaris.

Es important remarcar que l'aplicació de gestió d'usuaris únicament és funcional si l'executem a l'ordinador que conté també el servidor de Bases de Dades ja que al treballar amb la base de dades en mode de superusuari o root, per temes de seguretat no permet realitzar connexions remotes com a root.

La següent figura 27 mostra la pantalla d'entrada de l'administrador a l'aplicació de gestió d'usuaris.

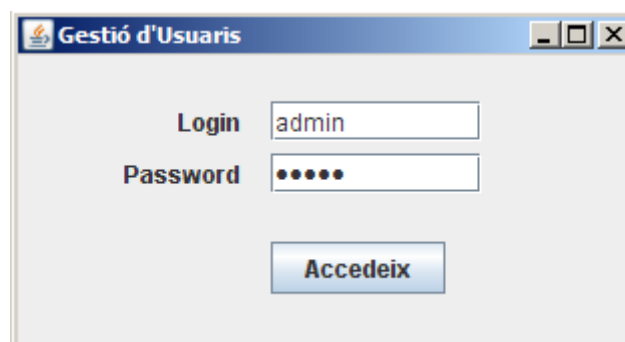


Figura 27: Pantalla d'entrada de l'administrador d'usuaris

Un cop introduïts el nom i el password d'administrador i si són correctes, s'accedeix a la pantalla que permet tant donar d'alta, com donar de baixa els usuaris desitjats.

Per realitzar l'operació simplement haurem d'introduir el nom i password del nou usuari així com el nom de la granja sobre la qual se li vol donar permisos per treballar, també s'haurà de seleccionar al desplegable d'operació si es vol fer una alta o una baixa .

El que farà l'aplicació internament serà accedir a la base de dades d'usuaris i crear el nou usuari amb la seva base de dades corresponent on posteriorment introduirà les dades de les granges.

La següent figura 28 mostra la pantalla on l'aplicació de gestió d'usuaris permet realitzar altes o baixes sobre els usuaris desitjats.

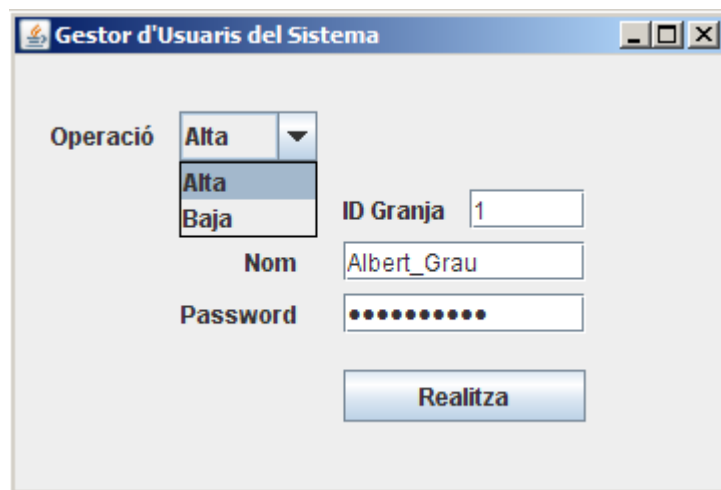


Figura 28: Pantalla d'alta/baixa d'usuaris

3.7.2 Funcionalitats compartides de l'aplicació web i de l'aplicació d'escriptori

Pràcticament totes les funcionalitats tan de l'aplicació web com de l'aplicació d'escriptori són les mateixes, raó per la qual les funcionalitats globals s'explicaran gairebé totes en aquest apartat.

El primer que es trobarà l'usuari a l'hora de voler accedir a l'aplicació serà la pantalla on se li demanarà el seu nom d'usuari i password, el qual per ser vàlid haurà d'haver estat creat prèviament amb l'aplicació de gestió d'usuaris.

La següent figura 29 mostra la pantalla de l'aplicació ASERP web on es demana el nom i password d'usuari (durant el període de desenvolupament de l'aplicació).

Aquesta imatge mostra una pantalla web d'autenticació. A la part superior, hi ha un capçalera gris amb el títol "Autenticació" en blanc. A sota, el fons és blanc. Hi ha dos camps d'entrada: "Nom" i "Password", cadascun amb un rectangle blanc per escriure. A sota d'aquests camps, hi ha un botó rectangular amb el text "Button".

Figura 29: Pantalla d'autenticació d'usuaris de l'aplicació web

Un cop l'usuari s'haurà autenticat correctament, accedirà al menú principal de l'aplicació, on se li mostraran les funcionalitats disponibles a l'aplicació. Aquestes funcionalitats seran les següents:

- Pujar fitxers
- Realitzar càlculs numèrics
- Realitzar càlculs gràfics

La següent figura 30 mostra la pantalla de l'aplicació ASERP web on es mostra el menú principal de l'aplicació web.

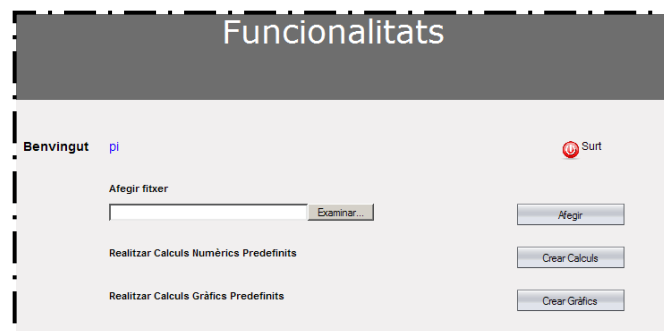
Aquesta imatge mostra la pantalla principal de l'aplicació. A la part superior, hi ha un capçalera gris amb el títol "Funcionalitats" en blanc. A sota, el fons és blanc. A l'esquerra, hi ha el text "Benvingut" seguit d'un petit icona "pi". A la dreta, hi ha un icona "Sur" amb un símbol de sortida. A sota, hi ha tres seccions: "Afegeix fitxer" amb un camp d'entrada i un botó "Examinar..."; "Realitzar Càlculs Numèrics Predefinits" amb un botó "Crear Càlculs"; i "Realitzar Càlculs Gràfics Predefinits" amb un botó "Crear Gràfics".

Figura 30: Pantalla de menú principal de l'aplicació web

En els següents apartats es descriuran acuradament les funcionalitats anteriorment llistades.

Funcionalitat de pujar fitxers

Quan un usuari es troba ja donat d'alta, el primer que necessita és una eina que li permeti llegir el seu fitxer de dades VSI i introduir les dades a la seva base de dades.

El que realitza l'aplicació quan l'usuari tria un fitxer es pujar el fitxer al servidor(en el cas de l'aplicació web), llegir el fitxer línia a línia i anar introduint les dades a les taules de la base de dades corresponents, segons l'especificació del format de fitxers VSI.

A més, l'aplicació permet que un usuari pugui pujar un o varis fitxers de dades de forma que aquestos són també introduïts correctament a la Base de Dades.

Així per exemple un usuari pot disposar d'un fitxer VSI amb dades de les seves granges entre els anys 1990 i 2000 i un altre amb dades compreses entre el 2000 i el 2007, de forma que pot pujar els dos fitxers si li resulta interessant per a poder treballar amb les dades conjuntament.

La següent figura 31 mostra la pantalla de l'aplicació ASERP web on es mostra com es pujaria un fitxer de dades VSI.

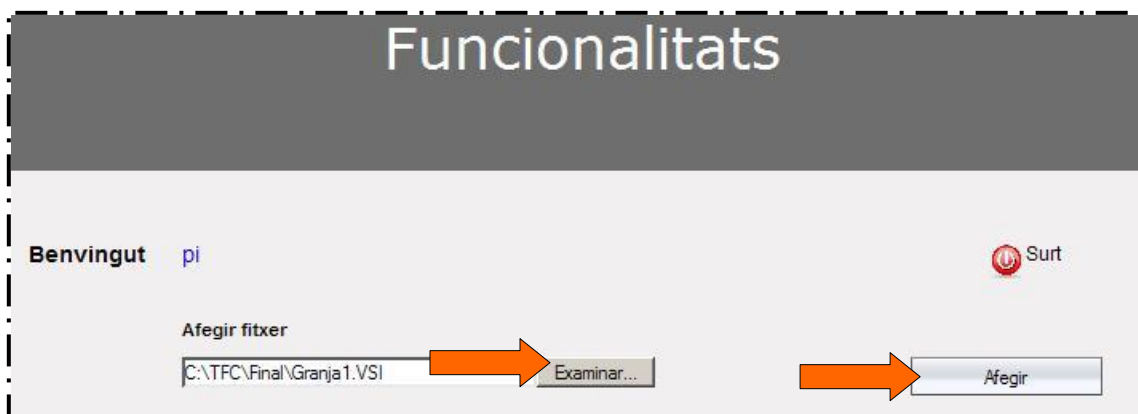


Figura 31: Exemple pujar fitxer VSI

Funcionalitat de Càlculs Numèrics Predefinits

La funcionalitat de Càlculs Numèrics Predefinits és la de permetre realitzar una sèrie de càlculs complexos sobre les dades disponibles dels animals de les granges.

L'acotació dels càlculs es fa mitjançant una data d'inici i una data de fi del període sobre el qual es volen fer els càlculs i el nom de la granja sobre la que volem treballar, ja que un fitxer VSI pot emmagatzemar dades d'una o de varies granges.

La següent figura 32 mostra com s'acotaria el període i la granja per a realitzar els càlculs.

The screenshot shows a window titled "Calculs Numèrics". Inside, there is a section "Defineix un període" with input fields for "Inici" (25/02/1999) and "Fi" (10/03/2000). Below these is a "Granja" field with the value "1". A "Permisos" field shows "tots". A calendar pop-up is open, showing the month of March 2000, with the date 10 selected. A "Calcular" button is located at the bottom right.

Figura 32: Acotació paràmetres per realitzar càlculs

Els càlculs que s'obtenen, es poden classificar en dos grans grups, el primer grup estaria format pels càlculs que donen com a resultat la mitjana dels resultats obtinguts per cada animal durant el període especificat. En tots aquests càlculs apart de proporcionar el valor de la mitjana, també es mostren la variança i la desviació típica.

El segon grup seria el format pels càlculs que ens donen resultats totals, és a dir per la seva naturalesa no te sentit donar el resultat com a un valor mig en el període. Un exemple seria el càlcul del nombre de truges disponibles en un període, el que voldríem no és la mitja sinó la quantitat de truges totals que tindríem.

Seguidament es descriurà detalladament cada càlcul un a un, primerament es descriuran els càlculs que donen com a resultat valors mitjans i posteriorment descriurem els càlculs que donen resultats totals.

Càlculs que donen com a resultat una mitja	
Garrins Deslletats	Càlcul de la mitjana de garrins que han estat deslletats durant un període determinat.
Edat Deslletament	Càlcul de la mitjana d'edat dels garrins en realitzar el deslletament.
Primeres Garrinades	Càlcul del nombre mig de garrins nascuts vius durant el primer part de l'animal.
IDCF	Interval mig entre el deslletament (cicle n-1) i una cubrició fèrtil que ha donat part (cicle n).
IDOE	Interval mig entre el deslletament (cicle n-1) i la primera cubrició (cicle n).
Nascuts Morts	Càlcul de la mitjana de garrins que naixen morts en un període determinat.
Garrins retirats	Càlcul de la mitjana del nombre de animals retirats després del part.
Edat Truja a la baixa	Determina mitjana de la edat de les truges al donar-les de baixa.
Edat Primera Cubrició	Càlcul de la mitjana d'edat d'una truja al realitzar el primer cobriment.
Edat Primer Part	Càlcul de la mitjana d'edat en el part de la truja d'una granja seleccionada y durant un període determinat.
Edat Truja al Part	Càlcul de la mitjana de l'edat de les truges al part.
Interval Entre Parts	Càlcul de la mitjana entre els diferents parts d'un mateix animal.
Nascuts Vius	Càlcul de la mitjana de garrins que naixen vius en un període determinat.

Taula 2: Càlculs d'ASERP que retornen una mitja

Càlculs que donen com a resultat un valor total	
Nombre de Truges	Calcula el nombre de femelles d'una granja concreta y que no hagi estat donada de baixa fora del període seleccionat.
Nº de Verros Presents	Calcula tots els animals mascles de la granja indicada i que encara no s'han donat de baixa o si se n'ha donat ha estat durant el període seleccionat.
Nº Truges per Verro	Calcula el nombre de truges disponibles per a cada verro durant el període seleccionat.
Nº truges de baixa	Nombre de truges diferents donades de baixa durant el període indicat, de la granja seleccionada
Garrins per Truja de Baixa	Garrins produïts durant la vida de les truges de baixa dividit pel nombre de truges de baixa.
Interval entre Deslletament i Baixa	Interval mig des del deslletament (quan es l'últim moviment registrat) a la eliminació de la truja.
Num. Deslletaments	Càlcul del nombre de deslletaments produïts en un període de temps determinat.
Deslletaments per Truja	Càlcul que indica el nombre de deslletaments dividit del nombre de truges productives.
Avortaments	Càlcul del nombre de parts avortats dividit dels parts totals; durant un període i granja determinats.
Prod. Truja Present	Aquest càlcul s'obté a partir de la divisió entre els deslletaments totals i el nombre de truges presents. El nombre de truges presents és la divisió entre els dies totals de presència de les verres i el període. El període és la divisió entre el nombre de cubricions fèrtils de les verres dividit per el total de cubricions, fèrtils o no fèrtils.
Prod. truja Productiva	Aquest càlcul s'obté a partir de la divisió entre els deslletaments totals i el nombre de truges productives. El nombre de truges productives és la divisió entre els dies totals de presència de les verres que han estat cobertes i el període. El període és la divisió entre el nombre de cubricions fèrtils de les verres dividit per el total de cubricions, fèrtils o no fèrtils.

Taula 3: Càlculs d'ASERP que retornen un resultat total.

La següent figura 33 mostra uns càlculs numèrics realitzats entre les dates 25/02/1999 i 10/03/2000 sobre la granja anomenada "1".

Defineix un període

Inici: 25/02/1999

Fi: 10/03/2000

mm/dd/yyyy

Granja: 1

Permisos: tots

Calcular

	Mitjana	Variança	Desviació Típica
Garrins Deslletats	11,02	3,39	1,84
Edat Desllet	21,03	12,46	3,53
Primeres Garrinades	11,26	9,77	3,13
IDCF	25,52	35.475,1	188,35
IDOE	17,07	35.878,59	189,42
Nascuts vius	11,76	10,29	3,21
Nascuts morts	0,98	2,28	1,51
Garrins retirats	0,64	1,75	1,32
Edat Truja a la baixa	793,6	155.437,14	394,26
Edat Primera Cubrició	255,62	10.618,76	103,05
Edat Primer Part	366,86	10.408,68	102,02
Edat Truja al Part	638,98	93.790,02	306,25
Interval Entre Parts	146,03	10.561,01	102,77

Figura 33: Exemple càlculs numèrics

Funcionalitat de Càlculs Gràfics Predefinits

La funcionalitat de Càlculs Gràfics, ens permet realitzar uns càlculs que podríem dir que són de menys complexitat que el Càlculs Numèrics descrits en l'apartat anterior i mostrar-los en forma de gràfica.

En aquests cas podem dividir els càlculs en quatre grans grups, els grups serien: gràfics per Interval, gràfics per Cicle, gràfics per Dia i gràfics per Setmana.

Seguidament veurem cada tipus de gràfic en detall i sobre quins càlculs sens permet fer cada tipus de gràfic.

- **Gràfics per Interval:** En aquest tipus de gràfic els intervals es mostren en dies, la informació d'un gràfic d'aquest tipus ens dirà el nombre d'animals que per cada interval que existeixi en el període especificat compleixen el càlcul demanat. En aquestos gràfics s'ha decidit que l'aplicació ASERP els mostri com a gràfics de barres 3D.

Els càlculs que tenim disponibles per realitzar aquest tipus de gràfics són:

- **Deslletament:** Mostra el nombre de garrins deslletats durant cada interval.
- **IDCF:** Mostra el nombre de repeticions per a cada Interval entre el deslletament i la cubrició fèrtil durant el període seleccionat.

La següent figura 34 mostra un gràfic per interval, dels deslletament produïts en una granja durant el període 01/01/2000 i 01/02/2000 sobre la granja anomenada "1".

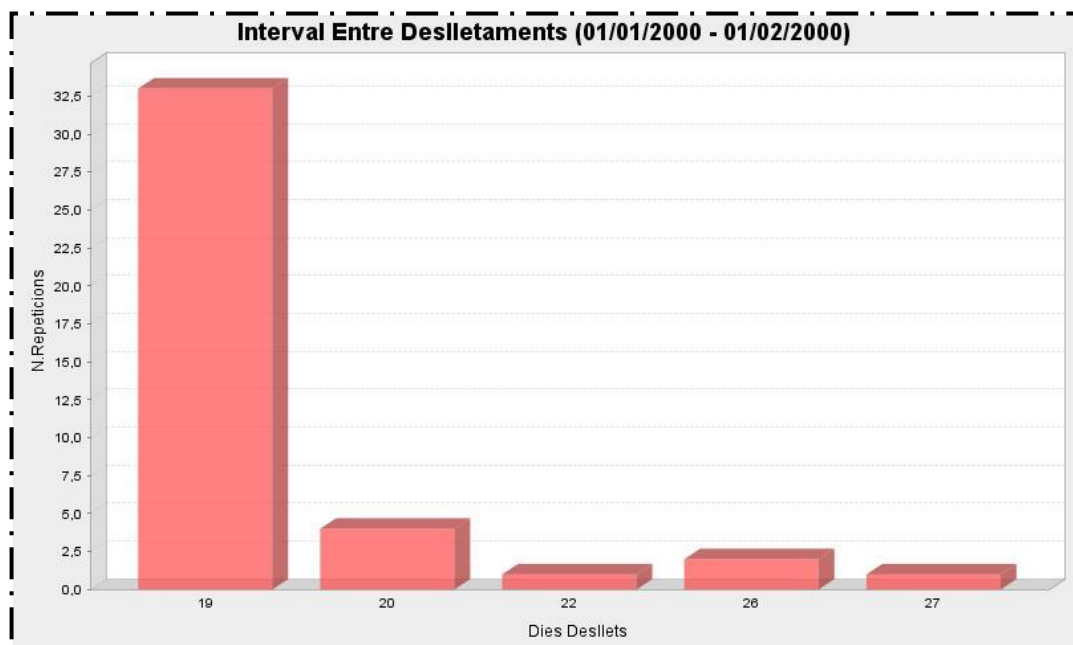


Figura 34: Exemple gràfic per interval

- **Gràfics per Cicle:** En aquest tipus de gràfic el que es mostra es el resultat total del càlcul demanat per a cada cicle reproductiu. En aquestos gràfics s'ha decidit que l'aplicació ASERP els mostri com a gràfics lineals en 3D.

Els càlculs que tenim disponibles per realitzar aquest tipus de gràfics són:

- **Numero d'Animals:** Mostra el nombre d'animals totals per cicle.
- **Deslletaments:** Mostra el nombre de garrins deslletats per cicle.
- **Nascuts Vius:** Mostra el nombre de nats vius per cicle.

La següent figura 35 mostra un gràfic per cicle, dels garrins nascuts vius en una granja durant el període 01/01/2000 i 01/02/2000 sobre la granja anomenada "1".

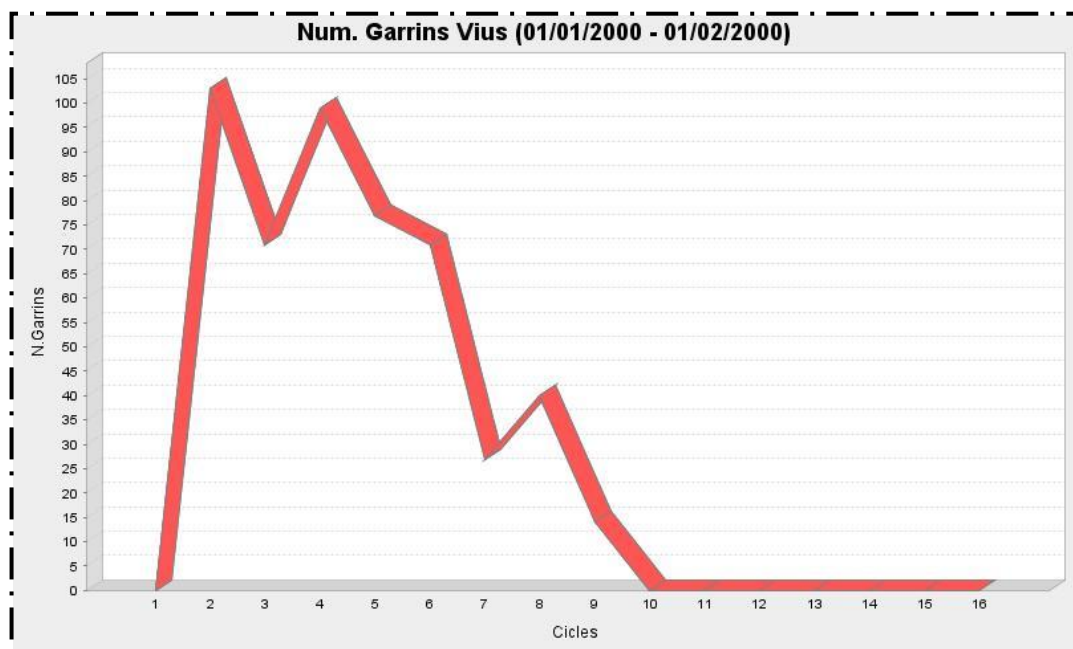


Figura 35: Exemple gràfic per cicle

- **Gràfics per Dia:** En aquest tipus de gràfic el que es mostra es el resultat del càlcul demanat per dia. En aquestos gràfics s'ha decidit que l'aplicació ASERP els mostri com a gràfics lineals.

Els càlculs que tenim disponibles per realitzar aquest tipus de gràfics són:

- **Nascuts vius:** Mostra el nombre de nats vius per dia.
- **Nascuts morts:** Mostra el nombre de nats morts per dia.
- **Num. Parts:** Mostra el nombre de parts per dia.
- **Num. Desllets:** Mostra el nombre de deslletats per dia.

La següent figura 36 mostra un gràfic per dia dels garrins nascuts vius en una granja durant el període 01/01/2000 i 01/02/2000 sobre la granja anomenada "1".

Observem que al realitzar un gràfic per dies d'un període de temps, la part inferior del gràfic mostra els dies del període seleccionat, en aquest cas com que s'ha triat un interval de 1 mes es veuen 31 dies.

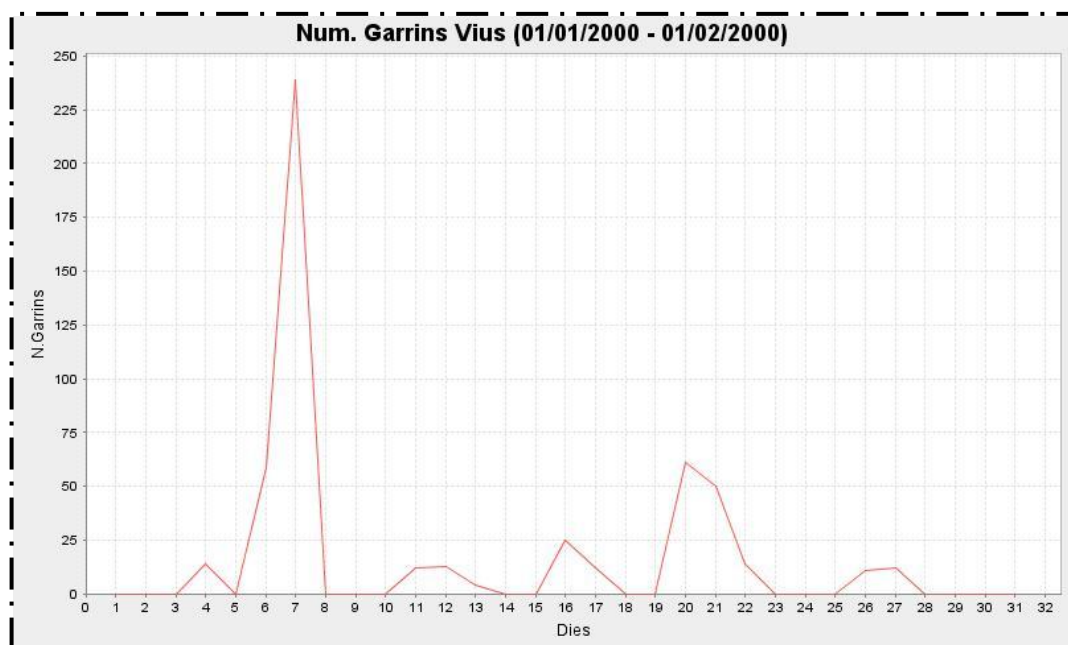


Figura 36: Exemple gràfic per dia

- **Gràfics per Setmana:** En aquest tipus de gràfic el que es mostra és el resultat del càlcul demanat per setmana. En aquestos gràfics s'ha decidit que l'aplicació ASERP els mostri com a gràfics lineals 3D.

Els càlculs que tenim disponibles per realitzar aquest tipus de gràfics són:

- **Garrins retirats:** Mostra el nombre de garrins retirats per setmana.
- **Nascuts vius:** Mostra el nombre de nats vius per setmana.
- **Nascuts morts:** Mostra el nombre de nats morts per setmana.
- **Garrins adoptats:** Mostra el nombre de garrins adoptats per setmana.
- **Num. Desllets:** Mostra el nombre de deslletaments per setmana.
- **Num. Parts:** Mostra el nombre de parts per setmana.
- **Num. Cubricions:** Mostra el nombre de cubricions per setmana.

La següent figura 37 mostra un gràfic per setmana dels garrins nascuts vius en una granja durant el període 01/01/2000 i 01/02/2000 sobre la granja anomenada "1".

Observem que al realitzar un gràfic per setmanes d'un període de temps, la part inferior del gràfic mostra les setmanes del període seleccionat, en aquest cas com que s'ha triat un interval de 1 mes es veuen 4 setmanes.

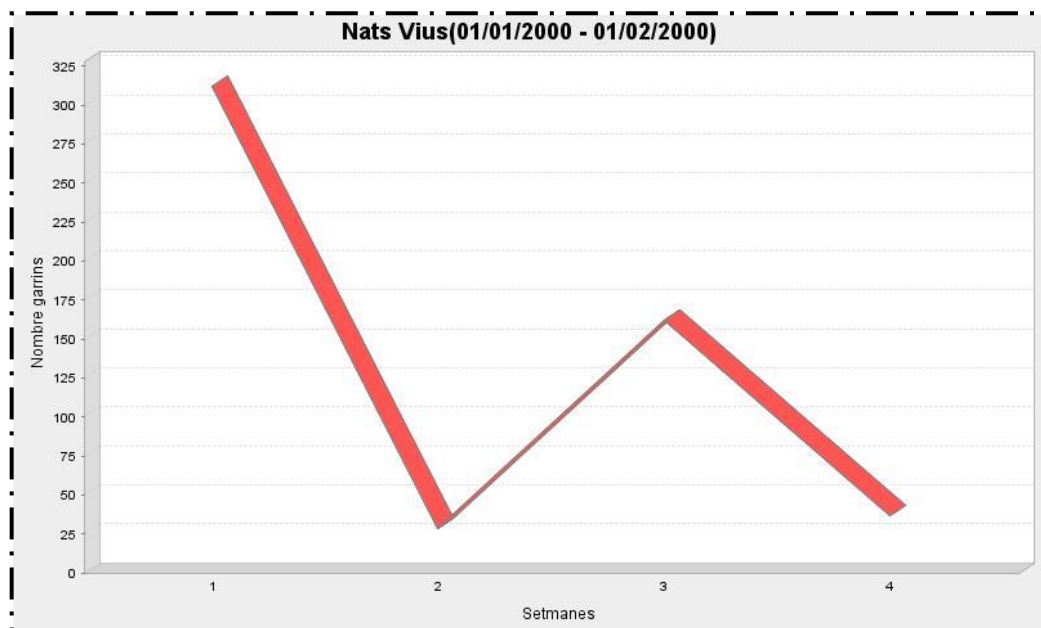


Figura 37: Exemple gràfic per setmana

Aspectes Concrets de l'aplicació ASERP d'escriptori

L'aplicació ASERP d'escriptori ofereix una funcionalitat afegida que és la de realitzar consultes simples sobre les taules Altas i Reproductivos de la base de dades .

Podríem descriure la funcionalitat com un gestor gràfic per a consultes sobre les taules, aquest gestor permet especificar els camps que volem filtrar, tant sigui un camp o més. Finalment al camp “*Valor a obtenir*” podem seleccionar quines columnes de la taula de la base de dades volem que ens mostrin al resultat de la consulta.

En el següent exemple la figura 38, mostra una consulta sobre la taula Altas on es selecciona que volem mostrar la fecb (Data de Baixa), que compleixi que la granja sigui 1 i que el sexe del animal sigui Mascle i el resultat d'aquesta.

The screenshot displays the ASERP application interface. The main window is titled 'Altas' and contains a form for specifying query criteria. The 'Valor a obtenir:' dropdown is set to 'fecb'. The form includes fields for 'granja' (set to 1), 'any', 'Ordre', 'Nº Animal', 'Sexe' (set to 'Mascle'), 'Raó', 'Raça', 'Baixa', 'FECN', 'FECE', 'FECB', and 'Origen'. A 'Realitzar Consulta' button is located at the bottom of the form. An orange arrow points from this button to a smaller window titled 'Resultat de la Consulta'. This window shows a list of dates under the column 'fecb'.

fecb
01-ene-1970
01-ene-1970
01-ene-1970
01-ene-1970
01-ene-1970
01-ene-1970
01-ene-1970
01-ene-1970
01-ene-2003
01-ene-2003
01-ene-2003
01-ene-1970
02-nov-1995
01-ene-1970
01-ene-2003
31-dic-1994
01-ene-2003

Figura 38: Consulta genèrica sobre la taula altas i resultat d'aquesta

Aspectes Concrets de l'aplicació ASERP Web

Com a aspectes concrets de les funcionalitats web, podem citar dos importants, el primer és la selecció d'idioma que se'ns deixa triar a la pàgina d'entrada a l'Aplicació.

La figura 39 mostra la pantalla de selecció d'idioma de l'aplicació web.



Figura 39: Pantalla de selecció d'idioma de l'aplicació web

El segon aspecte a comentar és que a la pàgina de funcionalitats principals de l'aplicació, disposem d'una funcionalitat anomenada "Surt", la qual té la finalitat de desconnectar-nos de la base de dades i sortir amb seguretat de l'espai d'usuari a la web.

3.8 Proves Realitzades

3.8.1 Configuracions

L'estructura de l'aplicació requereix tenir almenys un usuari donat d'alta per a poder treballar amb ella; és per això que ApliServer serà el primer programa que s'utilitzarà per part de l'administrador amb el fi de crear usuaris per a que aquests puguin treballar.

Cal tenir en compte que totes les aplicacions realitzades tenen com a requeriment tenir instal·lat els següents components:

Taula Requeriments	
ApliServer	
Client	Servidor
No aplicable	MySQL, Java (v5 o posterior)
ASERP vEscriptori	
Client	Servidor
Java (v5 o posterior)	MySQL, Java (v5 o posterior)
ASERP Web	
Client	Servidor
Navegador Web	Apache Tomcat, MySQL, Java (v5 o posterior)

Taula 4: Requeriments del Software realitzat.

Java

Per a la compilació dels projectes que formen l'aplicació, serà necessari tenir instal·lat al sistema el Java SE Development Kit (JDK).

Per a l'execució disposant dels fitxers binaris prèviament compilats simplement es requereix Java Runtime Environment (JRE).

MySQL

S'ha de tenir instal·lat quelcom versió d'aquesta base de dades al servidor.

Durant el procés de proves, la configuració que teníem ha estat la de per defecte.

Tant bon just s'ha instal·lat MySQL és necessari la creació de la BD anomenada Principal i la creació de la taula Users dins d'aquesta mateixa BD; per a que l'ApliServer pugui gestionar correctament les altes i les baixes d'usuaris de l'ASERP.

El codi corresponent a la creació de la BD principal, junt a la taula Users ve indicada per la figura següent:

```
create database principal;  
use principal;  
create table users (nom VARCHAR(15),password VARCHAR(15),  
idgranja VARCHAR(5));
```

Figura 40: Codi inicialització de la BD.

Apache Tomcat

El servidor de l'aplicació Web requereix la prèvia instal·lació d'un servidor Web. En el nostre cas hem elegit Apache Tomcat. Les raons ja han estat explicades anteriorment.

L'Apache Tomcat serà l'encarregat de posar en funcionament (o desplegament) l'aplicació Web ASERP, a partir del codi compilat i empaquetat en el fitxer WAR.

Aquest fitxer WAR, en el seu interior tindrà els fitxers JSP, les llibreries necessàries pel seu correcte funcionament de l'aplicació, junt amb les classes del paquet domini que aportaran tota la funcionalitat a l'aplicació.

3.8.2 Marc de proves, resultats i temps d'execució

Les proves s'han realitzat tant en entorn d'escriptori Windows com en Linux.

En les taules següents s'il·lustra els resultats obtinguts, en les diferents proves.

Client en Linux (Des de que sol·licita resultats)	Servidor Linux (Temps de resposta) **
Càlculs Numèrics (Període 1 any)	536ms
Càlculs Numèrics (Període 2 anys)	540ms
Càlculs Numèrics (Període 5 anys)	773ms
Càlculs Numèrics (Període 10 anys)	1357ms
Gràfics per cicle (Període 1 any)	134ms
Gràfics per cicle (Període 2 anys)	190ms
Gràfics per cicle (Període 5 anys)	353ms
Gràfics per cicle (Període 10 anys)	691ms
Inserció fitxer VSI al sistema	11'61 segons!!

Taula 5: Proves d'ASERP en un entorn Linux

Client en Windows (Des de que sol·licita resultats)	Servidor en Windows (Temps de resposta) **
Càlculs Numèrics (Període 1 any)	457ms
Càlculs Numèrics (Període 2 anys)	561ms
Càlculs Numèrics (Període 5 anys)	751ms
Càlculs Numèrics (Període 10 anys)	1242ms
Gràfics per cicle (Període 1 any)	291ms
Gràfics per cicle (Període 2 anys)	361ms
Gràfics per cicle (Període 5 anys)	370ms
Gràfics per cicle (Període 10 anys)	601ms
Inserció fitxer VSI al sistema	12'13 minuts!!

Taula 6: Proves d'ASERP en un entorn Windows

** El període tractat ha estat del 01/01/1991 al 01/01/2001, sobre un usuari que se l'hi ha carregat un fitxer amb 1489 registres en la taula Altas i 7195 en la taula reproductivos. El servidor en els dos casos era un PentiumM1.7 amb 1GB DDR a 333Mhz.

Capítol 4

Objectius Complerts i Treball Futur

Els objectius marcats a l' inici del TFC, han estat complerts en la seva totalitat, solvantant els problemes que han anat sorgint al llarg de tot el desenvolupament d'aquest i aprenent sobre algunes mancances de coneixements sobre part de les eines amb les quals s'ha treballat.

A més a més, s'ha realitzat una interfície d'escriptori de la mateixa aplicació ASERP per tal de poder treballar localment amb ella amb la finalitat de proporcionar una eina més ràpida i no haver de dependre d'Internet.

Per a aquesta versió d'escriptori se l'hi ha afegit la funcionalitat de realitzar quelcom consultes a la base de dades, estiguin o no prefixades.

Pel que fa al treball futur, seria interessant partint dels resultats obtinguts de l'aplicació ASERP creada en aquest TFC; poder realitzar estudis de viabilitat de les granges combinant els resultats que ja mostra l'aplicació, amb resultats econòmics de la mateixa granja sobre els mateixos períodes de temps. D'aquesta manera poder extreure conclusions per a la presa de decisions en la granja.

Un altre punt a destacar seria la substitució del format de fitxer VSI, per un format estàndard definit en llenguatge XML.

Capítol 5

Conclusió

Durant el transcurs d'aquest projecte s'han desenvolupat varis components de software que estableixen un model de programació d'aplicacions de simulacions en aquest cas ramaderes.

El sistema obtingut és lliure, modular, portable, escalable i segur. Així mateix l'aplicació ha donat suport a diferents entorns de treball per a que sigui l'usuari el que elegeixi treballar en un únic PC, utilitzar una xarxa d'ordinadors o treballar en línia. Aquest és un punt molt important avui en dia, ja que la tecnologia avança pel camí de la mobilitat

Durant la seva elaboració s'han tingut en compte les tècniques de programació modular, orientada a objectes, procurant que el resultat sigui una programació genèrica i orientada a aspectes.

Un aspecte a tenir en compte, és que l'aplicació final ha soltat totes les necessitats inicials, tot i que un cop finalitzada hem trobat que per l'estudi i presa de decisions es queda limitat. Aquest seria un segon pas en el que es podria utilitzar el potencial de l'aplicació en crear més operacions i agrupar gràfics, juntament amb dades financeres de l'explotació que ens puguin aportar més informació per la presa de decisions del l'empresari ramader.

El fet d'haver estat treballant amb un format de fitxers lineal com és el cas de VSI ha fet que haguéssim de fer algunes filigranes en el codi, que haguessin estat més senzilles i possiblement més efectives que les actuals; però al estar limitats ha haver de treballar amb aquest format s'ha tingut que dissenyar d'aquesta manera.

També creiem oportú realitzar una llicència GPL pel fet de que aquest codi no quedi en l'oblit d'un projecte final de carrera.

Ja per finalitzar, podem assegurar que la Aplicació sofrint unes petites adaptacions en el codi, es podria portar la seva funcionalitat en altres àmbits com podrien ser altres tipus de explotacions.

Al nostre parer, l'aplicació ofereix una gran quantitat de possibilitats, les quals vindrien limitades per la creativitat del propi programador.

Capítol 6

Llicència

El programari lliure és, abans de res, una forma ètica d'entendre el programari o els programes d'ordinador. Això inclou tant la fabricació dels programes com la seva distribució i la seva utilització. Per tant, el programari lliure no està orientat (ni és patrimoni exclusiu) únicament a la gent capaç d'escriure un programa, o a la gent que posseeix empreses i per tant està en condicions de fer negoci amb els programes d'ordinador. Per contra, el programari lliure té a veure amb l'usuari tant com amb els desenvolupadors o els comerciants. O tal vegada fins i tot més, ja que és la llibertat dels usuaris l'únic objectiu del moviment del programari lliure.

Així que el programari lliure com concepte i el moviment que ho fa créixer i evolucionar és un concepte (i un moviment) ètic. Molta gent es confon i considera el moviment del programari lliure com un moviment eminentment tècnic. Mentre que això últim pot ser cert per a altres moviments similars (per exemple, el moviment Open Source) no ho és en absolut quan vam parlar de programari lliure.

Però, què és, exactament, el programari lliure?

La definició mantinguda per la Fundació per al Programari Lliure diu que perquè un programa d'ordinador sigui considerat programari lliure ha de respectar quatre drets o llibertats considerats com fonamentals per a l'usuari d'aquest programa:

- Els usuaris han de tenir dret a utilitzar el programa, sense restriccions, on vulgui, com vulgui i per al que vulgui.
- Els usuaris han de tenir dret a estudiar com funciona el programa i, si ho desitgen, a adaptar-lo a les seves necessitats específiques.
- Els usuaris han de tenir dret a distribuir còpies als seus amics, empleats, coneguts, i empleats, en fi, a qualsevol persona que desitgin.
- Els usuaris han de tenir dret a millorar el programa, publicar i distribuir les seves millores al públic (o a qui desitgin) de manera que més persones surtin beneficiades dels canvis.

Per tant, per tot l' anteriorment descrit i per tal d'afavorir la difusió d'aquesta aplicació i així poder millorar per part d'altres programadors, considerem oportú llicenciar aquesta aplicació com a software lliure, seguint la llicència GPL (General Public License) de GNU.

Acrònims

API	Application Programming Interface
ASERP	Aplicació de Simulacions per l'Estudi Ramader Porcí
BBDD	Bases de Dades
BD	Base de Dades
BDC	Base de Dades Centralitzada
BDD	Base de Dades Distribuïda
CSS	Cascading Style Sheets
DTD	Document Type Definition
GCC	GNU Compiler Collection
GNU	GNU is Not Unix
GPL	General Public License
GUI	Graphical User Interface
HTTP	HyperText Transfer Protocol
IIS	Internet Information Services
ISP	Internet Service Provider
J2EE	Java 2 Enterprise Edition
J2ME	Java 2 Micro Edition
J2SE	Java 2 Standard Edition
JDBC	Java DataBase Connectivity
JDK	Java Development Kit
JFC	Java Foundation Classes
JNI	Java Native Interface
JSF	Java Server Faces
JSP	Java Server Pages
JVM	Java Virtual Machine
MVC	Model Vista Controlador
SGBD	Sistema Gestió de Base de Dades
SO	Sistema Operatiu
SQL	Structured Query Language
TCP	Transmission Control Protocol
UI	User Interface
W3C	World Wide Web Consortium
XML	eXtensible Markup Language

Glossari

[0]

Apache Tomcat

és un contenidor de servlets desenvolupat a l'Apache Software Foundation. Tomcat implementa les especificacions de servlet i de Java Server Pages (JSP) de Sun Microsystems, proporcionant un entorn per al codi Java a executar en cooperació amb un servidor web.

[1]

Applets

Un Applet és un component de programari que corre sobre el context d'un altre programa, per exemple, un navegador web. Els applets acostumen a desenvolupar una funció molt concreta que no pot ser usada de forma independent.

[2]

Col-lex

Gestor de Bases de Dades que utilitzar el llenguatge estructurat SQL.

[3]

Domini

Conjunt de classes que inclouen tota la funcionalitat d'una aplicació.

[4]

Firmware

Firmware és un bloc d'instruccions de programa per a propòsits específics, gravat en una memòria ROM, que estableix la lògica de més sota nivell que controla els circuits electrònics d'un dispositiu de qualsevol tipus. A l'estar integrat en l'electrònica del dispositiu és en part maquinari, però també és programari, ja que proporciona lògica i es disposa en algun tipus de llenguatge de programació.

[5]

Garbage Collector

El recol·lector d'escombraries (en anglès, garbage collector) és un mecanisme implícit de gestió de memòria implementat en alguns llenguatges de programació de tipus interpretat o semi-interpretat.

[6]

Innodb

InnoDB és una tecnologia d'emmagatzematge de dades de font oberta para MySQL, inclòs com format de taula estàndard en totes les distribucions de MySQL AB a partir de les versions 4.0. La seva característica principal és que suporta transaccions de tipus ACID i bloqueig de registres i integritat referencial.

[7]

Interfície

Conjunt de classes i components que s'encerreguen de proporcionar a l'usuari les eines necessàries per a poder interactuar gràficament amb les funcionalitats de l'aplicació.

[8]

JavaBeans

Els JavaBeans són un model de components creat per Sun Microsystems per al desenvolupament d'aplicacions en Java.

L'especificació de JavaBeans de Sun Microsystems els defineix com "components de programari reutilitzables que es puguin manipular visualment en una eina de desenvolupament".

[9]

Java Doc

És una utilitat de Sun Microsystems para generar APIs en format HTML d'un document de codi font Java. Javadoc és l'estàndard de la indústria per a documentar classes de Java. La majoria dels IDEs els generen automàticament.

[10]

JFreeChart

JFreeChart és una llibreria gratuïta per a la plataforma de programació Java, la funcionalitat bàsica d'aquesta llibreria és la de permetre realitzar gràfiques de molts tipus diferents en programes escrits en Java.

[11]

MyIsam

MyISAM és la tecnologia d'emmagatzematge de dades usada per defecte pel sistema administrador de bases de dades relacionals MySQL. Aquest tipus de taules estan basades en el format ISAM però amb noves extensions. En les últimes versions de Mysql, el motor InnoDB està començant a reemplaçar a aquest tipus de taules per la seva capacitat d'executar transaccions de tipus ACID i bloqueig de registres i integritat referencial.

[12]

MySQL

MySQL és un sistema de gestió de bases de dades relacional (anglès *RDBMS - Relational DataBase Management System*) multi-fil (*multithread*), multiusuari, que usa el llenguatge SQL (*Structured Query Language*).

[13]

OpenSource

El codi obert (*open source* en anglès) és habitualment programari, tot i que també pot ser maquinari, el codi font del qual està públicament disponible. Codi distribuït i desenvolupat lliurement, que ofereix a l'usuari la possibilitat d'entrar en el seu codi font per poder-lo estudiar i/o modificar.

[14]

Oracle

Oracle es un sistema de gestió de base de dades relacional (o RDBMS per l'acrònim en anglès de Relational Data Basi Management System), fabricat per Oracle Corporation.

[15]

Pilot-DB

El Pilot-DB, o abreviadament DB, és una base de dades relacional, com ara ThinkDB o JFile, de programari lliure, amb llicència GNU, disponible en català, pensada per a ordinadors de butxaca Palm OS.

[16]

PostgreSQL

PostgreSQL és un programari lliure que implementa un sistema de gestió de bases de dades objecte-relacional, distribuït sota llicència BSD.

[17]

Servlets

Els servlets són objectes Java executats per un servidor d'aplicacions i que responen a invocacions HTTP, servint pàgines dinàmiques.

El contingut generat pot ser un fitxer de qualsevol tipus, la majoria de vegades HTML.

[18]

Swing

Swing és una biblioteca gràfica para Java que forma part de les Java Foundation Classes (JFC). Inclou widgets per a interfície gràfica d'usuari tals com caixes de text, botons, desplegable i taules.

Bibliografia

- [ref-001] Hans Bergsten, *JavaServer Faces*. O'Reilly 2004.
- [ref-002] Jonas Jacobi, John R. Fallows, *Pro JSF and Ajax: Building Rich Internet Components*. Apress 2006.
- [ref-003] Budi Kuniawan, *Java for the Web with Servlets, JSP, and EJB : a developer's guide to J2EE solutions*, Indianapolis: New Riders Publishing 2002.
- [ref-004] Phil Hanna, *JSP manual de referencia*, McGraw-Hill Osborne corp. 2002.
- [ref-005] Randy Jay Yarger, George Reese, Tim King, *MySQL and mSQL*, Sebastopol, CA: O'Reilly, 1999.
- [ref-006] Paul Dubois, *MySQL*, Indianapolis, IN: New Riders, 1999.
- [ref-007] MySQL AB, *MySQL administrator's guide*, Indianapolis, Ind: Sams Pub., corp.2004
- [ref-008] Paul J. Perrone, *Building Java Enterprise System with J2EE*, Indianapolis: Sams Publishing, 2000.
- [ref-009] Jim Keogh, *J2EE manual de referencia*, McGraw-Hill corp. 2003.
- [ref-010] Donald Bales, *Java programming with Oracle JDBC*, Sebastopol: O'Reilly, 2002.

- [ref-011] <http://3tres3.com>
(Pàgina d'informació porcina).
- [ref-012] <http://www.doxygen.org>
(Documentació de la API)
- [ref-013] <http://www.object-refinery.com/jfreechart>
(Portal de la llibreria JFreeChart).
- [ref-014] <http://www.lawebdelprogramador.com>
(Forum on vam intercanviar informació sobre JSF i MySQL).
- [ref-015] <http://dev.mysql.com>
(Portal de desenvolupament amb MySQL)

Annexos:

Annex A:

Taules SQL

Annex B:

Documentació de l'ASERP

Annex C:

IDE Netbeans

Annex D:

Llicència GPL

Annex A

Taules SQL

Taula creada en la base de dades anomenada "Principal". Aquesta serà la taula on aniran emmagatzemats tots els usuaris de l'aplicació ASERP junt amb l'identificador de la granja que se l'hi ha assignat per a poder treballar i el seu password per entrar a l'aplicació.

```
CREATE TABLE Users (
    nom VARCHAR(15),
    password VARCHAR(15),
    idgranja VARCHAR(5)
);
```

Les dues taules següents són comuns en cadascuna de les bases de dades que s'han creat pels usuaris de l'aplicació ASERP. La descripció dels camps que les formen la podem trobar en l'interior del cos d'aquesta memòria; concretament en l'apartat Descripció i format del fitxer VSI.

```
CREATE TABLE Altas (
    id int,
    granja VARCHAR(5),
    any VARCHAR(1),
    orden VARCHAR(1),
    nanimal VARCHAR(4),
    ciclo VARCHAR(2),
    sexo VARCHAR(1),
    raza VARCHAR(2),
    fecn DATETIME,
    origen VARCHAR(5),
    fece DATETIME,
    baja VARCHAR(1),
    fecb DATETIME,
    razon VARCHAR(2),
    PRIMARY KEY(id)
);
```



```
CREATE TABLE Reproductivos (  
    id int,  
    granja VARCHAR(5),  
    any VARCHAR(1),  
    orden VARCHAR(1),  
    nanimal VARCHAR(4),  
    ciclo VARCHAR(2),  
    rep VARCHAR(1),  
    fechacub1 DATETIME,  
    fechacub2 DATETIME,  
    fechacub3 DATETIME,  
    fechacub4 DATETIME,  
    fechacub5 DATETIME,  
    macho1 VARCHAR(6),  
    macho2 VARCHAR(6),  
    macho3 VARCHAR(6),  
    macho4 VARCHAR(6),  
    macho5 VARCHAR(6),  
    tipocubr VARCHAR(5),  
    fechaparto DATETIME,  
    tipo VARCHAR(1),  
    viv VARCHAR(2),  
    mue VARCHAR(2),  
    momi VARCHAR(1),  
    ado VARCHAR(2),  
    ret VARCHAR(2),  
    fechadestete DATETIME,  
    nlechones VARCHAR(2),  
    fechacubfert DATETIME,  
    numcub VARCHAR(1),  
    PRIMARY KEY(id)  
);
```

Degut al cost que pot tenir la demora de determinades consultes, quan les base de dades pugui tenir uns quants milers de registres, s'ha optat per crear índex en les taules per tal d'agilitzar el temps de resposta.

CREATE INDEX nanimal on Altas(nanimal)

CREATE INDEX nanimal on Reproductivos(nanimal)

Annex B

Documentació de l'ASERP

Documentació TFC Documentació de l'Espai de Noms

Paquet domain

Classes

- class [Calculs](#)
Classe que implementa tot el que es calculs numerics.
- class [CalculsGrafics](#)
Classe que implementa tots els que es calculs necessaris per als grafics.
- class [CercaGranja](#)
Classe que serveix per saber sobre quina o quines granjes pot treballar un usuari.
- class [IntroAlta](#)
Classe que gestiona l'entrada de dades a la taula Altas.
- class [IntroReproductivos](#)
Classe que gestiona l'entrada de dades a la taula Reproductivos.
- class [IntroToBD](#)
Classe per introduir les dades a la Base de Dades.
- class [LlegirDades](#)
Classe encarregada de la lectura de dades.
- class [Manegador](#)
[Manegador](#) Principal, Gestio d'usuaris, operacions, grafics.
- class [Operacions](#)
Classe que s'encarrega de proporcionar metodes per fer consultes SQL i obtenir els resultats.

Documentació TFC Documentació de les Classes

Referència de la Classe domain.Calculs

Descripció Detallada

Classe que implementa tot el que es calculs numerics.

Autor:

blu3

Definició a la línia 23 del fitxer Calculs.java.

Mètodes públics

- [Calculs](#) ([Operacions](#) ope, String database)
Constructora de la classe.
- String [ProdTrujaPresent](#) (String nom, String inici, String fi) throws Exception
Calcul de la productivitat de les trujes presents.
- String [ProdTrujaProductiva](#) (String nom, String inici, String fi) throws Exception
Calcul de la Productivitat de les trujes productives.

- String [GarrinDesll](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana de garrins que han estat deslletats durant un periode determinat.
- String [Vius](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana de garrins que naixen vius en un periode determinat.
- String [Morts](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana de garrins que naixen morts en un periode determinat.
- String [Retiraments](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana del nombre de animals retirats despres del part.
- String [Deslletaments](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre de deslletaments produïts en un periode de temps determinat.
- String [Avortaments](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre de parts abortats / dels parts totals; durant un periode i granja determinats.
- String [GarriDesllPerTruja](#) (String nom, String inici, String fi) throws Exception
Calcul que indica el nombre de deslletaments / nombre de trujes productives.
- String [IntervalEntreParts](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana entre els diferents parts d'un mateix animal.
- String [EdatDeslletament](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana d'edat dels garrins en realitzar el deslletament.
- String [NombrePrimeresGarrinades](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre mig de garrins nascuts vius durant el primer part de l'animal.
- String [EdatPrimerCobrimet](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana d'edat d'una truja al realitzar el primer cobrimet.
- String [EdatPrimerPart](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana d'edat en el part de la truja d'una granja seleccionada y durant un periode determinat.
- String [EdatTrujesPart](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana de la edat de les trujes al part.
- String [NombreTrujesBaixa](#) (String nom, String inici, String fi) throws Exception
Nombre de trujes diferents donades de baixa durant el periode indicat, de la granja seleccionada.
- String [EdatTrujesBaixa](#) (String nom, String inici, String fi) throws Exception
Determina mitjana de la edat de les trujes al donar-les de baixa.
- String [NombreTrujes](#) (String nom, String inici, String fi) throws Exception
Calcula el num de femelles de una granja concreta y que no hagi estat donada de baixa fora del periode seleccionat.
- String [NombreBerrosPresents](#) (String nom, String inici, String fi) throws Exception
Calcula tots els animals mascles de la granja indicada i que encara no s'han donat de baixa o si se n'ha donat ha estat durant el periode seleccionat.
- String [TrujesPerBerro](#) (String nom, String inici, String fi) throws Exception

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

- String [IDCF](#) (String nom, String inici, String fi) throws Exception
IDCF: Interval mig entre el desllet (cicle n-1) i una cubricio fertil que ha donat part (cicle n).
- String [IDOE](#) (String nom, String inici, String fi) throws Exception
IDOE: Interval mig entre el desllet (cicle n-1) i la primera cubricio (cicle n).
- String [NombreGarriPerTrujesBaixa](#) (String nom, String inici, String fi) throws Exception
NombreGarriPerTrujesBaixa: garrins produïts en la seva vida les trujes de baixa dividit pel nombre de truges de baixa.

- **String [IDBB](#) (String nom, String inici, String fi) throws Exception**
IDBB: Interval mig des del desllet (quan es l'últim moviment registrat) a la eliminació de la truja.
- **String [VariancaIDOE](#) ()**
Calcul de la variància IDOE, realitzant en el seu calcul principal.
- **String [VariancaIDBB](#) ()**
Calcul de la variància IDBB, realitzant en el seu calcul principal.
- **String [VariancaIDCF](#) ()**
Calcul de la variància IDCF, realitzant en el seu calcul principal.
- **String [VariancaETB](#) ()**
Calcul de la variància ETB, realitzant en el seu calcul principal.
- **String [VariancaGarrinDesll](#) ()**
Calcul de la variància dels garrins deslletats, realitzant en el seu calcul principal.
- **String [VariancaETP](#) ()**
Calcul de la variància ETP, realitzant en el seu calcul principal.
- **String [VariancaEPP](#) ()**
Calcul de la variància EPP, realitzant en el seu calcul principal.
- **String [VariancaE1C](#) ()**
Calcul de la variància E1C, realitzant en el seu calcul principal.
- **String [VariancaNPG](#) ()**
Calcul de la variància NPG, realitzant en el seu calcul principal.
- **String [VariancaED](#) ()**
Calcul de la variància ED, realitzant en el seu calcul principal.
- **String [VariancaNL](#) ()**
Calcul de la variància NL, realitzant en el seu calcul principal.
- **String [VariancaPTPdate](#) ()**
Calcul de la variància PTPdate, realitzant en el seu calcul principal.
- **String [VariancaPTPROD](#) ()**
Calcul de la variància PTPROD, realitzant en el seu calcul principal.
- **String [VariancaVIUS](#) ()**
Calcul de la variància vius, realitzant en el seu calcul principal.
- **String [VariancaMorts](#) ()**
Calcul de la variància morts, realitzant en el seu calcul principal.
- **String [VariancaRetirats](#) ()**
Calcul de la variància Retirats, realitzant en el seu calcul principal.
- **String [VariancaleP](#) ()**
Calcul de la variància leP, realitzant en el seu calcul principal.
- **String [stdIDOE](#) ()**
Calcul de la desviació típica IDOE, realitzant en el seu calcul principal.
- **String [stdIDBB](#) ()**
Calcul de la desviació típica IDBB, realitzant en el seu calcul principal.
- **String [stdIDCF](#) ()**
Calcul de la desviació típica IDCF, realitzant en el seu calcul principal.
- **String [stdETB](#) ()**
Calcul de la desviació típica ETB, realitzant en el seu calcul principal.
- **String [stdETP](#) ()**
Calcul de la desviació típica ETP, realitzant en el seu calcul principal.
- **String [stdEPP](#) ()**
Calcul de la desviació típica EPP, realitzant en el seu calcul principal.
- **String [stdE1C](#) ()**
Calcul de la desviació típica E1C, realitzant en el seu calcul principal.
- **String [stdGarrinDesll](#) ()**
Calcul de la desviació típica dels garrins deslletats, realitzant en el seu calcul principal.

- **String [stdNPG](#) ()**
Calcul de la desviació típica NPG, realitzant en el seu calcul principal.
- **String [stdED](#) ()**
Calcul de la desviació típica ED, realitzant en el seu calcul principal.
- **String [stdNL](#) ()**
Calcul de la desviació típica NL, realitzant en el seu calcul principal.
- **String [stdPTPdate](#) ()**
Calcul de la desviació típica PTPdate, realitzant en el seu calcul principal.
- **String [stdPTPROD](#) ()**
Calcul de la desviació típica PTPPROD, realitzant en el seu calcul principal.
- **String [stdVIUS](#) ()**
Calcul de la desviació típica vius, realitzant en el seu calcul principal.
- **String [stdMorts](#) ()**
Calcul de la desviació típica morts, realitzant en el seu calcul principal.
- **String [stdRetirats](#) ()**
Calcul de la desviació típica Retirats, realitzant en el seu calcul principal.
- **String [stdleP](#) ()**
Calcul de la desviació típica leP, realitzant en el seu calcul principal.

Documentació del Constructor i el Destructor

domain.Calculs.Calculs ([Operacions](#) ope, String *database*)

Constructora de la classe.

Paràmetres:

ope Instància d'Operacions que ens servira per realitzar els calculs

database String que conte el nom de la BBDD assignada a l'usuari

Definició a la línia 42 del fitxer Calculs.java.

Referències domain.Calculs.db i domain.Calculs.op.

Documentació de les Funcions Membre

String domain.Calculs.ProdTrujaPresent (String *nom*, String *inici*, String *fi*) throws

Exception

Calcul de la productivitat de les trujes presents.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 60 del fitxer Calculs.java.

String domain.Calculs.ProdTrujaProductiva (String *nom*, String *inici*, String *fi*) throws

Exception

Calcul de la Productivitat de les trujes productives.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 120 del fitxer `Calculs.java`.

String domain.Calculs.GarrinDesII (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana de garrins que han estat deslletats durant un període determinat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 158 del fitxer `Calculs.java`.

String domain.Calculs.Vius (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana de garrins que naixen vius en un període determinat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 187 del fitxer `Calculs.java`.

String domain.Calculs.Morts (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana de garrins que naixen morts en un període determinat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 217 del fitxer `Calculs.java`.

String domain.Calculs.Retiraments (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana del nombre de animals retirats després del part.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 247 del fitxer `Calculs.java`.

String domain.Calculs.Deslletaments (String nom, String inici, String fi) throws Exception

Calcul del nombre de deslletaments produïts en un període de temps determinat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 277 del fitxer Calculs.java.

String domain.Calculs.Avortaments (String nom, String inici, String fi) throws Exception

Calcul del nombre de parts abortats / dels parts totals; durant un període i granja determinats.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 302 del fitxer Calculs.java.

String domain.Calculs.GarriDeslPerTruja (String nom, String inici, String fi) throws Exception

Calcul que indica el nombre de deslletaments / nombre de trujes productives.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

String domain.Calculs.IntervalEntreParts (String nom, String inici, String fi) throws Exception

Calcul de la mitjana entre els diferents parts d'un mateix animal.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 359 del fitxer Calculs.java.

String domain.Calculs.EdatDeslletament (String nom, String inici, String fi) throws Exception

Calcul de la mitjana d'edat dels garrins en realitzar el deslletament.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 389 del fitxer `Calculs.java`.

String domain.Calculs.NombrePrimeresGarrinades (String *nom*, String *inici*, String *fi*) throws Exception

Calcul del nombre mig de garrins nascuts vius durant el primer part de l'animal.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 419 del fitxer `Calculs.java`.

String domain.Calculs.EdatPrimerCobrimient (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana d'edat d'una truja al realitzar el primer cobrimient.

Tenint en compte una granja seleccionada i un periode determinat

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 450 del fitxer `Calculs.java`.

String domain.Calculs.EdatPrimerPart (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana d'edat en el part de la truja d'una granja seleccionada y durant un periode determinat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 480 del fitxer `Calculs.java`.

String domain.Calculs.EdatTrujesPart (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana de la edat de les trujes al part.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 511 del fitxer `Calculs.java`.

String domain.Calculs.NombreTrujesBaixa (String *nom*, String *inici*, String *fi*) throws Exception

Nombre de trujes diferents donades de baixa durant el periode indicat, de la granja seleccionada.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 541 del fitxer `Calculs.java`.

String [domain.Calculs.EdatTrujesBaixa](#) (String *nom*, String *inici*, String *fi*) throws Exception

Determina mitjana de la edat de les trujes al donar-les de baixa.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 575 del fitxer `Calculs.java`.

String domain.Calculs.NombreTrujes (String *nom*, String *inici*, String *fi*) throws Exception

Calcula el num de femelles de una granja concreta y que no hagi estat donada de baixa fora del periode seleccionat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 592 del fitxer `Calculs.java`.

String domain.Calculs.NombreBerrosPresents (String *nom*, String *inici*, String *fi*) throws Exception

Calcula tots els animals mascles de la granja indicada i que encara no s'han donat de baixa o si se n'ha donat ha estat durant el periode seleccionat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del periode que volem estudiar
fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 618 del fitxer `Calculs.java`.

String domain.Calculs.TrujesPerBerro (String *nom*, String *inici*, String *fi*) throws Exception

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

Paràmetres:

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 641 del fitxer `Calculs.java`.

Referències `domain.Calculs.nberros`.

Referenciat a `domain.Manegador.MTrujesPerBerro()`.

String domain.Calculs.IDCF (String *nom*, String *inici*, String *fi*) throws Exception

IDCF: Interval mig entre el desllet (cicle n-1) i una cubricio fertil que ha donat part (cicle n).

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 658 del fitxer `Calculs.java`.

String domain.Calculs.IDOE (String *nom*, String *inici*, String *fi*) throws Exception

IDOE: Interval mig entre el desllet (cicle n-1) i la primera cubricio (cicle n).

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 690 del fitxer `Calculs.java`.

String domain.Calculs.NombreGarriPerTrujesBaixa (String *nom*, String *inici*, String *fi*) throws Exception

NombreGarriPerTrujesBaixa: garrins produïts en la seva vida les trujes de baixa dividit pel nombre de truges de baixa.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 721 del fitxer `Calculs.java`.

String domain.Calculs.IDBB (String *nom*, String *inici*, String *fi*) throws Exception

IDBB:Interval mig des del desllet (quan es l'ultim moviment registrat) a la eliminacio de la truja.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 758 del fitxer `Calculs.java`.

String domain.Calculs.VariancaIDOE ()

Calcul de la variancia IDOE, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia

Definició a la línia 801 del fitxer `Calculs.java`.

String domain.Calculs.VariancaIDBB ()

Calcul de la variancia IDBB, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia

Definició a la línia 813 del fitxer `Calculs.java`.

String domain.Calculs.VariancaIDCF ()

Calcul de la variancia IDCF, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia

Definició a la línia 826 del fitxer `Calculs.java`.

String domain.Calculs.VariancaETB ()

Calcul de la variancia ETB, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia

Definició a la línia 839 del fitxer `Calculs.java`.

String domain.Calculs.VariancaGarrinDesII ()

Calcul de la variancia dels garrins deslletats, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 852 del fitxer Calculs.java.

String domain.Calculs.VariancaETP ()

Calcul de la variancia ETP, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 865 del fitxer Calculs.java.

String domain.Calculs.VariancaEPP ()

Calcul de la variancia EPP, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 878 del fitxer Calculs.java.

String domain.Calculs.VariancaE1C ()

Calcul de la variancia E1C, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 891 del fitxer Calculs.java.

String domain.Calculs.VariancaNPG ()

Calcul de la variancia NPG, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 903 del fitxer Calculs.java.

String domain.Calculs.VariancaED ()

Calcul de la variancia ED, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 916 del fitxer Calculs.java.

String domain.Calculs.VariancaNL ()

Calcul de la variancia NL, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 929 del fitxer Calculs.java.

String domain.Calculs.VariancaPTPdate ()

Calcul de la variancia PTPdate, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 941 del fitxer Calculs.java.

String domain.Calculs.VariancaPTPROD ()

Calcul de la variancia PTPROD, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia

Definició a la línia 954 del fitxer `Calculs.java`.

String `domain.Calculs.VariancaVIUS ()`

Calcul de la variancia vius, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia

Definició a la línia 968 del fitxer `Calculs.java`.

String `domain.Calculs.VariancaMorts ()`

Calcul de la variancia morts, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia

Definició a la línia 981 del fitxer `Calculs.java`.

String `domain.Calculs.VariancaRetirats ()`

Calcul de la variancia Retirats, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia

Definició a la línia 994 del fitxer `Calculs.java`.

String `domain.Calculs.VariancaleP ()`

Calcul de la variancia leP, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia

Definició a la línia 1006 del fitxer `Calculs.java`.

String `domain.Calculs.stdIDOE ()`

Calcul de la desviació típica IDOE, realitzant en el seu calcul principal.

Retorna:

El resultat de la std

Definició a la línia 1019 del fitxer `Calculs.java`.

String `domain.Calculs.stdIDBB ()`

Calcul de la desviació típica IDBB, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica

Definició a la línia 1031 del fitxer `Calculs.java`.

String `domain.Calculs.stdIDCF ()`

Calcul de la desviació típica IDCF, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica

Definició a la línia 1044 del fitxer `Calculs.java`.

String `domain.Calculs.stdETB ()`

Calcul de la desviació típica ETB, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica

Definició a la línia 1057 del fitxer `Calculs.java`.

String `domain.Calculs.stdETP ()`

Calcul de la desviació típica ETP, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica

Definició a la línia 1070 del fitxer `Calculs.java`.

String [domain.Calculs.stdEPP \(\)](#)

Calcul de la desviació típica EPP, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1083 del fitxer Calculs.java.

String [domain.Calculs.stdE1C \(\)](#)

Calcul de la desviació típica E1C, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1096 del fitxer Calculs.java.

String [domain.Calculs.stdGarrinDesII \(\)](#)

Calcul de la desviació típica dels garrins deslletats, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1108 del fitxer Calculs.java.

String [domain.Calculs.stdNPG \(\)](#)

Calcul de la desviació típica NPG, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1120 del fitxer Calculs.java.

String [domain.Calculs.stdED \(\)](#)

Calcul de la desviació típica ED, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1133 del fitxer Calculs.java.

String [domain.Calculs.stdNL \(\)](#)

Calcul de la desviació típica NL, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1146 del fitxer Calculs.java.

String [domain.Calculs.stdPTPdate \(\)](#)

Calcul de la desviació típica PTPdate, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1158 del fitxer Calculs.java.

String [domain.Calculs.stdPTPROD \(\)](#)

Calcul de la desviació típica PTPROD, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1171 del fitxer Calculs.java.

String [domain.Calculs.stdVIUS \(\)](#)

Calcul de la desviació típica vius, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1185 del fitxer Calculs.java.

String [domain.Calculs.stdMorts \(\)](#)

Calcul de la desviació típica morts, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1198 del fitxer `Calculs.java`.

String domain.Calculs.stdRetirats ()

Calcul de la desviació típica Retirats, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1211 del fitxer `Calculs.java`.

String domain.Calculs.stdleP ()

Calcul de la desviació típica leP, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 1223 del fitxer `Calculs.java`.

Referència de la Classe domain.CalculsGrafics

Descripció Detallada

Classe que implementa tots els que es calculs necessaris per als grafics.

Autor:

admin

Definició a la línia 25 del fitxer `CalculsGrafics.java`.

Mètodes públics

- [CalculsGrafics](#) ([Operacions](#) ope, String database)
Constructora de la classe.
- String [CorreccioData](#) (Calendar data)
Mètode que transforma una data de tipus Calendar a un String en format ddMMyyyy.
- List [ViusSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul dels de garrins que naixen vius en un període determinat per setmanes.
- List [MortsSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul de la garrins que naixen morts en un període determinat per setmanes.
- List [RetiramentsSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul dels animals retirats després del part per setmanes.
- List [NumPartsSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul dels parts produïts en el període indicat per setmanes.
- List [NCubSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre de cubricions produïdes en el període indicat per setmanes.
- List [AdoptatsSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre de garrins adoptats en el període indicat per setmanes.
- List [NDeslletamentsSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre de deslletaments produïts en el període indicat per setmanes.
- List [NDeslletamentsDiaGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre de deslletaments produïts en el període indicat per dies.
- List [NPartsDiaGrafic](#) (String nom, String inici, String fi) throws Exception

Calcul del nombre de parts produïts en el periode indicat per dies.

- List [NViusDiaGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre de garrins nascuts vius en el periode indicat per dies.
- List [NMortsDiaGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre de garrins nascuts morts en el periode indicat per dies.
- List [NViusCicleGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre de garrins nascuts vius en el periode indicat per cicle.
- List [NDeslletatsCicleGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre de garrins deslletats en el periode indicat per cicle.
- List [NAnimalsCicleGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre d'animals en el periode indicat per cicle.
- List [IntervalDeslletamentGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul del interval entre el desllet i un nou part dels animals.
- List [IntervalIDCFGrafic](#) (String nom, String inici, String fi) throws Exception
Calcul del interval entre el desllet i la cubricio fertil dels animals.

Documentació del Constructor i el Destructor

domain.CalculsGrafics.CalculsGrafics ([Operacions](#) ope, String database)

Constructora de la classe.

Paràmetres:

ope Instancia d'Operacions que ens servira per realitzar els calculs
database String que conte el nom de la BBDD assignada a l'usuari
Definició a la línia 42 del fitxer CalculsGrafics.java.

Documentació de les Funcions Membre

String domain.CalculsGrafics.CorreccioData (Calendar data)

Metode que transforma una data de tipus Calendar a un String en format ddMMyyyy.

Paràmetres:

data Calendar que conte la data a transformar

Retorna:

String amb la data transformada en format ddMMyyyy
Definició a la línia 57 del fitxer CalculsGrafics.java.

List domain.CalculsGrafics.ViusSetmanaGrafic (String nom, String inici, String fi) throws Exception

Calcul dels de garrins que naixen vius en un periode determinat per setmanes.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del periode que volem estudiar
fi Indica el fi del periode que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul per setmanes
Definició a la línia 88 del fitxer *CalculsGrafics.java*.

**List domain.CalculsGrafics.MortsSetmanaGrafic (String *nom*, String *inici*, String *fi*)
throws Exception**

Calcul de la garrins que naixen morts en un periode determinat per setmanes.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del periode que volem estudiar
fi Indica el fi del periode que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul per setmanes
Definició a la línia 132 del fitxer *CalculsGrafics.java*.

**List domain.CalculsGrafics.RetiramentsSetmanaGrafic (String *nom*, String *inici*, String *fi*)
throws Exception**

Calcul dels animals retirats despres del part per setmanes.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del periode que volem estudiar
fi Indica el fi del periode que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul per setmanes
Definició a la línia 176 del fitxer *CalculsGrafics.java*.

**List domain.CalculsGrafics.NumPartsSetmanaGrafic (String *nom*, String *inici*, String *fi*)
throws Exception**

Calcul dels parts produïts en el periode indicat per setmanes.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del periode que volem estudiar
fi Indica el fi del periode que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul per setmanes
Definició a la línia 220 del fitxer `CalculsGrafics.java`.

List domain.CalculsGrafics.NCubSetmanaGrafic (String nom, String inici, String fi) throws Exception

Calcul del nombre de cubricions produïdes en el període indicat per setmanes.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del període que volem estudiar
fi Indica el fi del període que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul per setmanes
Definició a la línia 264 del fitxer `CalculsGrafics.java`.

List domain.CalculsGrafics.AdoptatsSetmanaGrafic (String nom, String inici, String fi) throws Exception

Calcul del nombre de garrins adoptats en el període indicat per setmanes.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del període que volem estudiar
fi Indica el fi del període que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul per setmanes
Definició a la línia 307 del fitxer `CalculsGrafics.java`.

List domain.CalculsGrafics.NDeslletamentsSetmanaGrafic (String nom, String inici, String fi) throws Exception

Calcul del nombre de deslletaments produïts en el període indicat per setmanes.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del període que volem estudiar
fi Indica el fi del període que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul en setmanes

Definició a la línia 350 del fitxer `CalculsGrafics.java`.

List domain.CalculsGrafics.NDeslletamentsDiaGrafic (String nom, String inici, String fi) throws Exception

Calcul del nombre de deslletaments produïts en el període indicat per dies.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul en dies

List domain.CalculsGrafics.NPartsDiaGrafic (String nom, String inici, String fi) throws Exception

Calcul del nombre de parts produïts en el període indicat per dies.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul en dies

Definició a la línia 440 del fitxer `CalculsGrafics.java`.

List domain.CalculsGrafics.NViusDiaGrafic (String nom, String inici, String fi) throws Exception

Calcul del nombre de garrins nascuts vius en el període indicat per dies.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul en dies

Definició a la línia 483 del fitxer `CalculsGrafics.java`.

List domain.CalculsGrafics.NMortsDiaGrafic (String *nom*, String *inici*, String *fi*) throws Exception

Calcul del nombre de garrins nascuts morts en el periode indicat per dies.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul en dies

Definició a la línia 525 del fitxer CalculsGrafics.java.

List domain.CalculsGrafics.NViusCicleGrafic (String *nom*, String *inici*, String *fi*) throws Exception

Calcul del nombre de garrins nascuts vius en el periode indicat per cicle.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul en cicles

List domain.CalculsGrafics.NDeslletatsCicleGrafic (String *nom*, String *inici*, String *fi*) throws Exception

Calcul del nombre de garrins deslletats en el periode indicat per cicle.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul en cicles

Definició a la línia 598 del fitxer CalculsGrafics.java.

List domain.CalculsGrafics.NAnimalsCicleGrafic (String *nom*, String *inici*, String *fi*) throws Exception

Calcul del nombre d'animals en el periode indicat per cicle.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del periode que volem estudiar
fi Indica el fi del periode que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul en cicles
 Definició a la línia 626 del fitxer *CalculsGrafics.java*.

List domain.CalculsGrafics.IntervalDeslletamentGrafic (String *nom*, String *inici*, String *fi*) throws Exception

Calcul del interval entre el desllet i un nou part dels animals.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del periode que volem estudiar
fi Indica el fi del periode que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul seguint el patro: interval, numero aparicions de l'interval
 Definició a la línia 658 del fitxer *CalculsGrafics.java*.

List domain.CalculsGrafics.IntervalIDCFGrafic (String *nom*, String *inici*, String *fi*) throws Exception

Calcul del interval entre el desllet i la cubricio fertil dels animals.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del periode que volem estudiar
fi Indica el fi del periode que volem estudiar

Excepcions:

java.lang.Exception

Retorna:

List llista amb els resultats del calcul seguint el patro: interval, numero aparicions de l'interval

Referència de la Classe domain.CercaGranja

Descripció Detallada

Classe que serveix per saber sobre quina o quines granjes pot treballar un usuari.

Autor:

blu3

Definició a la línia 22 del fitxer CercaGranja.java.

Mètodes públics

- [CercaGranja](#) (String user) throws Exception
Creates a new instance of [CercaGranja](#).
- void [CrearConnexio](#) () throws Exception
Metode encarregat de crear la connexio amb la Base de Dades.
- void [TancarConnexio](#) () throws SQLException
Metode encarregat de tancar la connexio amb la Base de Dades un cop finalitzada.
- String [Cerca](#) () throws Exception
Metode que realitza la cerca de la granja disponible.

Atributs Privats

- String [url](#) = ("jdbc:mysql://localhost/principal")
- Connection [conn](#) = null
- [Operacions ope](#)
- String [usuari](#)

Documentació del Constructor i el Destructor

domain.CercaGranja.CercaGranja (String user) throws Exception

Creates a new instance of [CercaGranja](#).

Paràmetres:

user String que correspon a l'usuari sobre el que volem cercar la granja disponible

Excepcions:

java.lang.Exception

Definició a la línia 34 del fitxer CercaGranja.java.

Documentació de les Funcions Membre

void domain.CercaGranja.CrearConnexio () throws Exception

Metode encarregat de crear la connexio amb la Base de Dades.

Excepcions:

java.lang.Exception

Definició a la línia 43 del fitxer CercaGranja.java.

void domain.CercaGranja.TancarConnexio () throws SQLException

Metode encarregat de tancar la connexio amb la Base de Dades un cop finalitzada.

Excepcions:

java.sql.SQLException

Definició a la línia 53 del fitxer CercaGranja.java.

String domain.CercaGranja.Cerca () throws Exception

Metode que realitza la cerca de la granja disponible.

Excepcions:*java.lang.Exception***Retorna:**

String amb la granja disponible
 Definició a la línia 63 del fitxer CercaGranja.java.

Referència de la Classe domain.IntroAlta

Descripció Detallada

Classe que gestiona l'entrada de dades a la taula Altas.

Autor:

Admin

Definició a la línia 24 del fitxer IntroAlta.java.

Mètodes públics

- [IntroAlta](#) ()
Creates a new instance of [IntroAlta](#).
- void [tractarAlta](#) (Connection conn, String linea, String nom) throws SQLException, ParseException
Gestio i introduccio de les dades procedents del fitxer VSI cap a les variables corresponents de la taula Altas.
- void [InsereixData](#) (int index, int iniSubst, int fiSubst) throws ParseException, SQLException
Insereix totes les dates de la taula Altas.
- String [ComprovaDataBuida](#) (String data)
Comprova que la data estigui expressada correctament.

Documentació de les Funcions Membre

void domain.IntroAlta.tractarAlta (Connection conn, String linea, String nom) throws SQLException, ParseException

Gestio i introduccio de les dades procedents del fitxer VSI cap a les variables corresponents de la taula Altas.

Paràmetres:

conn Conector a la base de dades MySQL

linea String que conte una línia de text a tractar del fitxer VSI

nom String que indica el nom de l'usuari, o el que es el mateix, el nom de la BBDD

Definició a la línia 44 del fitxer IntroAlta.java.

void domain.IntroAlta.InsereixData (int *index*, int *iniSubst*, int *fiSubst*) throws ParseException, SQLException

Insereix totes les dates de la taula Altas.

Paràmetres:

index int que correspon al index on s'ha d'afegir dins de la taula

iniSubst int que indica la posició exacta de l'inici de l'String que s'ha de passar a la BBDD

fiSubst int que indica la posició exacta del final de l'String que s'ha de passar a la BBDD

Excepcions:

Definició a la línia 102 del fitxer IntroAlta.java.

Referències domain.IntroAlta.ComprovaDataBuida(), domain.IntroAlta.df, domain.IntroAlta.línia i domain.IntroAlta.pstmt.

String domain.IntroAlta.ComprovaDataBuida (String *data*)

Comprova que la data estigui expressada correctament.

Paràmetres:

data String que conte una data a analitzar o blancs

Retorna:

String amb la data final ja tractada i llesta per afegir a la taula corresponent de la BBDD

Excepcions:

Definició a la línia 119 del fitxer IntroAlta.java.

Referència de la Classe domain.IntroReproductivos

Descripció Detallada

Classe que gestiona l'entrada de dades a la taula Reproductivos.

Autor:

Admin

Definició a la línia 27 del fitxer IntroReproductivos.java.

Mètodes públics

- [IntroReproductivos \(\)](#)
Creates a new instance of [IntroReproductivos](#).
- void [tractarReproductivos](#) (Connection conn, String linea, String nom) throws SQLException, ParseException
Gestio i introduccio de les dades procedents del fitxer VSI cap a les variables corresponents de la taula Reproductivos.
- void [InsereixData](#) (int index, int iniSubst, int fiSubst) throws ParseException, SQLException

Insereix les dates a la taula Reproductivos (Excepte per les dates de Cubricio).

- void [InsereixDataCub](#) (int index, int iniSubst, int fiSubst) throws ParseException, SQLException

Insereix les dates de Cubricio a la taula Reproductivos i s'encarrega d'actualitzar la fechacubmax quan correspon.

- String [ComprovaDataBuida](#) (String data)
Comprova que la data estigui expressada correctament.

Documentació del Constructor i el Destructor

domain.IntroReproductivos.IntroReproductivos ()

Creates a new instance of [IntroReproductivos](#).

Definició a la línia 36 del fitxer IntroReproductivos.java.

Documentació de les Funcions Membre

void domain.IntroReproductivos.tractarReproductivos (Connection conn, String linea, String nom) throws SQLException, ParseException

Gestió i introducció de les dades procedents del fitxer VSI cap a les variables corresponents de la taula Reproductivos.

Paràmetres:

conn Conector a la base de dades MySQL

linea String que conte una línia de text a tractar del fitxer VSI

nom String que indica el nom de l'usuari, o el que es el mateix, el nom de la BBDD

Excepcions:

@throws java.sql.SQLException

java.text.ParseException

Definició a la línia 50 del fitxer IntroReproductivos.java.

Referències domain.IntroReproductivos.df, domain.IntroReproductivos.fechacubmax, domain.IntroReproductivos.numcub i domain.IntroReproductivos.pstmt.

void domain.IntroReproductivos.InsereixData (int index, int iniSubst, int fiSubst) throws ParseException, SQLException

Insereix les dates a la taula Reproductivos (Excepte per les dates de Cubricio).

Paràmetres:

index int que correspon al index on s'ha d'afegir dins de la taula

iniSubst int que indica la posició exacta de l'inici de l'String que s'ha de passar a la BBDD

fiSubst int que indica la posició exacta del final de l'String que s'ha de passar a la BBDD

Excepcions:

java.text.ParseException

java.sql.SQLException

Definició a la línia 124 del fitxer IntroReproductivos.java.

**void domain.IntroReproductivos.InsereixDataCub (int *index*, int *iniSubst*, int *fiSubst*)
throws ParseException, SQLException**

Insereix les les dates de Cubricio a la taula Reproductivos i s'encarrega d'actualitzar la fechacubmax quan correspon.

Paràmetres:

index int que correspon al index on s'ha d'afegir dins de la taula

iniSubst int que indica la posicio exacta de l'inici de l'String que s'ha de passar a la BBDD

fiSubst int que indica la posicio exacta del final de l'String que s'ha de passar a la BBDD

Excepcions:

java.text.ParseException

java.sql.SQLException

Definició a la línia 141 del fitxer IntroReproductivos.java.

String domain.IntroReproductivos.ComprovaDataBuida (String *data*)

Comprova que la data estigui expressada correctament.

Paràmetres:

data String que conte una data a analitzar o blancs

Retorna:

String amb la data final ja tractada i llesta per afegir a la taula corresponent de la BBDD

Excepcions:

Definició a la línia 160 del fitxer IntroReproductivos.java.

Referència de la Classe domain.IntroToBD

Descripció Detallada

Classe per introduir les dades a la Base de Dades.

Autor:

Admin

Definició a la línia 22 del fitxer IntroToBD.java.

Mètodes públics

- **[IntroToBD](#)** (Connection [conn](#), String [nom](#)) throws SQLException, ClassNotFoundException, InstantiationException, IllegalAccessException
Creates a new instance of [IntroToBD](#).
- **void [Tractarlinea](#)** (String *linea*) throws InstantiationException, IllegalAccessException, ParseException, SQLException, Exception

Metode que partint d'una linea amb format VSI, diferencia si es tracta d'una linea d'alta o de reproductivos i l'afegeix a la Base de Dades.

Documentació del Constructor i el Destructor

domain.IntroToBD.IntroToBD (Connection *conn*, String *nom*) throws SQLException, ClassNotFoundException, InstantiationException, IllegalAccessException

Creates a new instance of [IntroToBD](#).

Paràmetres:

conn Conector a la base de dades MySQL

nom String amb el nom de la base de dades sobre la que introduïrem les dades

Excepcions:

java.sql.SQLException

java.lang.ClassNotFoundException

java.lang.InstantiationException

java.lang.IllegalAccessException

Definició a la línia 39 del fitxer IntroToBD.java.

Documentació de les Funcions Membre

void domain.IntroToBD.Tractarlinea (String *linea*) throws InstantiationException, IllegalAccessException, ParseException, SQLException, Exception

Metode que partint d'una linea amb format VSI, diferencia si es tracta d'una linea d'alta o de reproductivos i l'afegeix a la Base de Dades.

Paràmetres:

linea String que conte una linea en format VSI

Excepcions:

java.lang.InstantiationException

java.lang.IllegalAccessException

java.text.ParseException

java.sql.SQLException

java.lang.Exception

Definició a la línia 56 del fitxer IntroToBD.java.

Referència de la Classe domain.LlegirDades

Descripció Detallada

Classe encarregada de la lectura de dades.

Autor:

Admin

Definició a la línia 21 del fitxer LlegirDades.java.

Mètodes públics

- [LlegirDades](#) (Connection conn, String nom, String fitxer) throws Exception
Lectura del fitxer seleccionat i obtenció de dades per a la bbdd.

Documentació del Constructor i el Destructor

domain.LlegirDades.LlegirDades (Connection *conn*, String *nom*, String *fitxer*) throws Exception

Lectura del fitxer seleccionat i obtenció de dades per a la bbdd.

Paràmetres:

conn Tipus Connection, enllac entre java i mysql
nom String que indica el nom d'usuari i la bbdd que te assignada
fitxer String que conte la ruta exacta d'on es troba el fitxer a llegir

Excepcions:

Exception Excepció general del sistema
 Definició a la línia 39 del fitxer LlegirDades.java.
 Referències domain.LlegirDades.bf, domain.LlegirDades.introbd,
 domain.LlegirDades.sCadena i domain.IntroToBD.TractarLinea().

Referència de la Classe domain.Manegador

Descripció Detallada

[Manegador](#) Principal, Gestió d'usuaris, operacions, gràfics.

Autor:

Admin

Definició a la línia 22 del fitxer Manegador.java.

Mètodes públics

- [Manegador](#) ()
Constructora del manegador del sistema.
- [Operacions IniOperacions](#) () throws Exception
Inicialització de les operacions.
- void [CreaConsulta](#) (String dataselect, String datafrom, String datawhere) throws Exception
Mètode a partir del qual es crea una consulta a una bbdd.
- void [LlegirDadesFitxer](#) (Connection [conn](#), String nom, String fitxer) throws Exception
Mètode encarregat de la lectura del fitxer i posterior càrrega de dades a la bbdd.
- void [AfegeixNovaCondicio](#) (String nova_condicio)
Introducció d'una o més condicions que han de complir les dades d'una consulta.
- ResultSet [ObteResultSet](#) (String usuari) throws SQLException
Obtenció del ResultSet de la consulta.
- List< String > [ObteLlistaResultats](#) ([Operacions ope](#), ResultSet rs, String columna) throws SQLException

Obte una llista resultats apartir d'un resultset.

- **[Operacions ObteOperacio \(\)](#)**
Obte l'instancia de l'operacio amb la qual s'esta treballant.
- **`void SeleccUsuari (String urluser, String usuari, String password)`**
Selecciona Usuari que ha entrat al sistema i emmagatzema la seva informacio.
- **`void CrearConnexio () throws Exception`**
Estableix una nova connexio amb la bbdd.
- **`void TancarConnexio () throws SQLException`**
Finalitza la connexio amb la bbdd.
- **`String ObteUsuari ()`**
Obtenir l'usuari actual del sistema.
- **`Connection obtenirConn ()`**
Obtenir l'instancia de la connexio.
- **`void InicialitzaCalculs ()`**
Inicialitza els calculs creant un objecte del tipus [Calculs](#) (numerics) i [CalculsGrafics](#).
- **`String MObtenirGranjaDisponible () throws Exception`**
Obtencio de la granja procedent de la base de dades principal.
- **`String MProdTrujaPresent (String nom, String inici, String fi) throws Exception`**
Calcul de la productivitat de les trujes presents.
- **`String MProdTrujaProductiva (String nom, String inici, String fi) throws Exception`**
Calcul de la Productivitat de les trujes productives.
- **`String MGarrinDesll (String nom, String inici, String fi) throws Exception`**
Calcul de la mitjana de garrins que han estat deslletats durant un periode determinat.
- **`String MViuis (String nom, String inici, String fi) throws Exception`**
Calcul de la mitjana de garrins que naixen vius en un periode determinat.
- **`String MMorts (String nom, String inici, String fi) throws Exception`**
Calcul de la mitjana de garrins que naixen morts en un periode determinat.
- **`String MDeslletaments (String nom, String inici, String fi) throws Exception`**
Calcul del nombre de deslletaments produïts en un periode de temps determinat.
- **`String MAvortaments (String nom, String inici, String fi) throws Exception`**
Calcul del nombre de parts abortats / dels parts totals; durant un periode i granja determinats.
- **`String MRetiraments (String nom, String inici, String fi) throws Exception`**
Calcul de la mitjana del nombre de animals retirats despres del part.
- **`String MGarriDesllPerTruja (String nom, String inici, String fi) throws Exception`**
Calcul que indica el nombre de deslletaments / nombre de trujes productives.
- **`String MIntervalEntreParts (String nom, String inici, String fi) throws Exception`**
Calcul de la mitjana entre els diferents parts d'un mateix animal.
- **`String MEdatDeslletament (String nom, String inici, String fi) throws Exception`**
Calcul de la mitjana d'edat dels garrins en realitzar el deslletament.
- **`String MIDCF (String nom, String inici, String fi) throws Exception`**
IDCF: Interval mig entre el desllet (cicle n-1) i una cubricio fertil que ha donat part (cicle n).
- **`String MIDOE (String nom, String inici, String fi) throws Exception`**
IDOE: Interval mig entre el desllet (cicle n-1) i la primera cubricio (cicle n).

- String [MNombrePrimeresGarrinades](#) (String nom, String inici, String fi) throws Exception
Calcul del nombre mig de garrins nascuts vius durant el primer part de l'animal.
- String [MEdatPrimerCobriment](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana d'edat d'una truja al realitzar el primer cobriment.
- String [MEdatPrimerPart](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana d'edat en el part de la truja d'una granja seleccionada y durant un periode determinat.
- String [MEdatTrujesPart](#) (String nom, String inici, String fi) throws Exception
Calcul de la mitjana de la edat de les trujes al part.
- String [MNombreTrujesBaixa](#) (String nom, String inici, String fi) throws Exception
Nombre de trujes diferents donades de baixa durant el periode indicat, de la granja seleccionada.
- String [MNombreGarriPerTrujesBaixa](#) (String nom, String inici, String fi) throws Exception
NombreGarriPerTrujesBaixa: garrins produïts en la seva vida les trujes de baixa dividit pel nombre de truges de baixa.
- String [MEdatTrujesBaixa](#) (String nom, String inici, String fi) throws Exception
Determina mitjana de la edat de les trujes al donar-les de baixa.
- String [MIDBB](#) (String nom, String inici, String fi) throws Exception
IDBB:Interval mig des del desllet (quan es l'ultim moviment registrat) a la eliminacio de la truja.
- String [MNombreTrujes](#) (String nom, String inici, String fi) throws Exception
Calcula el nun de femelles de una granja concreta y que no hagi estat donada de baixa fora del periode seleccionat.
- String [MNombreBerrosPresents](#) (String nom, String inici, String fi) throws Exception
Calcula tots els animals mascles de la granja indicada i que encara no s'han donat de baixa o si se n'ha donat ha estat durant el periode seleccionat.
- String [MTrujesPerBerro](#) (String nom, String inici, String fi) throws Exception

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

- String [stdIDOE](#) ()
Calcul de la desviacio tipica IDOE, realitzant en el seu calcul principal.
- String [stdIDBB](#) ()
Calcul de la desviacio tipica IDBB, realitzant en el seu calcul principal.
- String [stdIDCF](#) ()
Calcul de la desviacio tipica IDCF, realitzant en el seu calcul principal.
- String [stdETB](#) ()
Calcul de la desviacio tipica ETB, realitzant en el seu calcul principal.
- String [stdETP](#) ()
Calcul de la desviacio tipica ETP, realitzant en el seu calcul principal.
- String [stdEPP](#) ()
Calcul de la desviacio tipica EPP, realitzant en el seu calcul principal.
- String [stdGarrinDesll](#) ()
Calcul de la desviacio tipica dels garrins deslletats, realitzant en el seu calcul principal.

- **String [stdE1C](#) ()**
Calcul de la desviació típica E1C, realitzant en el seu calcul principal.
- **String [stdNPG](#) ()**
Calcul de la desviació típica NPG, realitzant en el seu calcul principal.
- **String [stdED](#) ()**
Calcul de la desviació típica ED, realitzant en el seu calcul principal.
- **String [stdNL](#) ()**
Calcul de la desviació típica NL, realitzant en el seu calcul principal.
- **String [stdPTPdate](#) ()**
Calcul de la desviació típica PTPdate, realitzant en el seu calcul principal.
- **String [stdPTPROD](#) ()**
Calcul de la desviació típica PTPROD, realitzant en el seu calcul principal.
- **String [stdVIUS](#) ()**
Calcul de la desviació típica vius, realitzant en el seu calcul principal.
- **String [stdMorts](#) ()**
Calcul de la desviació típica morts, realitzant en el seu calcul principal.
- **String [stdRetirats](#) ()**
Calcul de la desviació típica Retirats, realitzant en el seu calcul principal.
- **String [stdleP](#) ()**
Calcul de la desviació típica leP, realitzant en el seu calcul principal.
- **String [VariancaGarrinDesll](#) ()**
Calcul de la variància dels garrins deslletats, realitzant en el seu calcul principal.
- **String [VariancaIDOE](#) ()**
Calcul de la variància IDOE, realitzant en el seu calcul principal.
- **String [VariancaIDBB](#) ()**
Calcul de la variància IDBB, realitzant en el seu calcul principal.
- **String [VariancaIDCF](#) ()**
Calcul de la variància IDCF, realitzant en el seu calcul principal.
- **String [VariancaETB](#) ()**
Calcul de la variància ETB, realitzant en el seu calcul principal.
- **String [VariancaETP](#) ()**
Calcul de la variància ETP, realitzant en el seu calcul principal.
- **String [VariancaEPP](#) ()**
Calcul de la variància EPP, realitzant en el seu calcul principal.
- **String [VariancaE1C](#) ()**
Calcul de la variància E1C, realitzant en el seu calcul principal.
- **String [VariancaNPG](#) ()**
Calcul de la variància NPG, realitzant en el seu calcul principal.
- **String [VariancaED](#) ()**
Calcul de la variància ED, realitzant en el seu calcul principal.
- **String [VariancaNL](#) ()**
Calcul de la variància NL, realitzant en el seu calcul principal.
- **String [VariancaPTPdate](#) ()**
Calcul de la variància PTPdate, realitzant en el seu calcul principal.
- **String [VariancaPTPROD](#) ()**
Calcul de la variància PTPROD, realitzant en el seu calcul principal.

- String [VariancaVIUS \(\)](#)
Calcul de la variancia vius, realitzant en el seu calcul principal.
- String [VariancaMorts \(\)](#)
Calcul de la variancia morts, realitzant en el seu calcul principal.
- String [VariancaRetirats \(\)](#)
Calcul de la variancia Retirats, realitzant en el seu calcul principal.
- String [VariancaleP \(\)](#)
Calcul de la variancia leP, realitzant en el seu calcul principal.
- List [MViusSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MMortsSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MRetiradosSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MNumPartsSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MNDeslletamentsSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MAadoptatsSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MNCubSetmanaGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MNMortsDiaGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MNViusDiaGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MNPartsDiaGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MNDeslletamentsDiaGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MNViusCicleGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MNAanimalsCicleGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MNDeslletatsCicleGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MIntervalDeslletamentGrafic](#) (String nom, String inici, String fi) throws Exception
- List [MIntervalIDCFGrafic](#) (String nom, String inici, String fi) throws Exception

Documentació del Constructor i el Destructor

domain.Manegador.Manegador ()

Constructora del manegador del sistema.

Excepcions:

Definició a la línia 43 del fitxer Manegador.java.

Referències domain.Manegador.nf.

Documentació de les Funcions Membre

[Operacions](#) domain.Manegador.IniOperacions () throws Exception

Inicialització de les operacions.

Excepcions:

Definició a la línia 62 del fitxer Manegador.java.

Referències domain.Manegador.conn i domain.Manegador.ope.

void domain.Manegador.CreaConsulta (String *dataselect*, String *datafrom*, String *datawhere*) throws Exception

Metode a partir del qual es crea una consulta a una bbdd.

Paràmetres:

dataselect Camps resultat que volem obtenir
datafrom taules que volem consultar
datawhere condicions que volem que compleixin les dades

Excepcions:

Definició a la línia 77 del fitxer Manegador.java.

Referències domain.Operacions.CreaConsulta() i domain.Manegador.ope.

void domain.Manegador.LlegirDadesFitxer (Connection *conn*, String *nom*, String *fitxer*) throws Exception

Metode encarregat de la lectura del fitxer i posterior carrega de dades a la bbdd.

Paràmetres:

conn Conector entre la bbdd i l'aplicació
nom Identificador de la base de dades amb la que estem interactuant
fitxer Ruta exacta d'on es troba localitzat el fitxer d'entrada de dades (fitxer vsi)

Excepcions:

Definició a la línia 91 del fitxer Manegador.java.

Referències domain.Manegador.conn.

void domain.Manegador.AfegeixNovaCondicio (String *nova_condicio*)

Introducció d'una o més condicions que han de complir les dades d'una consulta.

Paràmetres:

nova_condicio Condició escrita en sql
 Definició a la línia 102 del fitxer Manegador.java.

ResultSet domain.Manegador.ObteResultSet (String *usuari*) throws SQLException

Obtenció del ResultSet de la consulta.

Paràmetres:

usuari Identificador de la bbdd amb la qual estem interactuant

Retorna:

ResultSet de la consulta

Excepcions:

Definició a la línia 115 del fitxer Manegador.java.

Referències domain.Operacions.ExeConsultaGenerica(),
 domain.Operacions.ObteLlistaQuery() i domain.Manegador.ope.

List<String> domain.Manegador.ObteLlistaResultats ([Operacions](#) ope, ResultSet rs, String *columna*) throws SQLException

Obte una llista resultats apartir d'un resultset.

Paràmetres:

ope Instrancia d'una operacio
rs ResultSet que volem analitzar
columna Identificador del resultat que volem extreure del ResultSet

Retorna:

Llista amb els resultats indicats en el camp "columna"

Excepcions:

Definició a la línia 130 del fitxer Manegador.java.
 Referències domain.Operacions.CreaLlistaResultats() i domain.Manegador.ope.

[Operacions](#) domain.Manegador.ObteOperacio ()

Obte l'instancia de l'operacio amb la qual s'esta treballant.

Retorna:

Instancia de l'operacio

Excepcions:

Definició a la línia 140 del fitxer Manegador.java.
 Referències domain.Manegador.ope.

void domain.Manegador.SelecUsuari (String *urluser*, String *usuari*, String *password*)

Selecciona Usuari que ha entrat al sistema i emmagatzema la seva informacio.

Paràmetres:

urluser URL de l'usuari per connectar amb la bbdd
usuari nom de l'usuari al sistema
password password de l'usuari

Excepcions:

Definició a la línia 155 del fitxer Manegador.java.

void domain.Manegador.CrearConnexio () throws Exception

Estableix una nova connexio amb la bbdd.

Excepcions:

Definició a la línia 166 del fitxer Manegador.java.
 Referències domain.Manegador.conn, domain.Manegador.connector,
 domain.Manegador.db, domain.Manegador.pasworddb i domain.Manegador.url.

void domain.Manegador.TancarConnexio () throws SQLException

Finalitza la connexio amb la bbdd.

Excepcions:

Definició a la línia 176 del fitxer Manegador.java.

Referències domain.Manegador.conn.

String domain.Manegador.ObteUsuari ()

Obtenir l'usuari actual del sistema.

Retorna:

nom de l'usuari actual del sistema

Excepcions:

Definició a la línia 186 del fitxer Manegador.java.

Referències domain.Manegador.db.

Referenciat a domain.Manegador.InicialitzaCalculs() i
domain.Manegador.MObtenirGranjaDisponible().

Connection domain.Manegador.obtenirConn ()

Obtenir l'instancia de la connexio.

Retorna:

Instancia de la connexio a la bbdd

Excepcions:

Definició a la línia 196 del fitxer Manegador.java.

Referències domain.Manegador.conn.

void domain.Manegador.InicialitzaCalculs ()

Inicialitza els calculs creant un objecte del tipus [Calculs](#) (numerics) i [CalculsGrafics](#).

Excepcions:

Definició a la línia 208 del fitxer Manegador.java.

Referències domain.Manegador.iniCalc, domain.Manegador.ObteUsuari() i
domain.Manegador.ope.

String domain.Manegador.MObtenirGranjaDisponible () throws Exception

Obtencio de la granja procedent de la base de dades principal.

Retorna:

Identificador de la granja corresponent

Excepcions:

Definició a la línia 223 del fitxer Manegador.java.

Referències domain.CercaGranja.Cerca() i domain.Manegador.ObteUsuari().

String domain.Manegador.MProdTrujaPresent (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la productivitat de les trujes presents.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 239 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i
domain.Calculs.ProdTrujaPresent().

String domain.Manegador.MProdTrujaProductiva (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la Productivitat de les trujes productives.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 255 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i
domain.Calculs.ProdTrujaProductiva().

String domain.Manegador.MGarrinDesII (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana de garrins que han estat deslletats durant un periode determinat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 271 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.GarrinDesll() i domain.Manegador.nf.

String domain.Manegador.MVius (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana de garrins que naixen vius en un període determinat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 287 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.Vius().

String domain.Manegador.MMorts (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana de garrins que naixen morts en un període determinat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 303 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.Morts() i domain.Manegador.nf.

String domain.Manegador.MDeslletaments (String *nom*, String *inici*, String *fi*) throws Exception

Calcul del nombre de deslletaments produïts en un període de temps determinat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 319 del fitxer Manegador.java.

Referències domain.Manegador.c i domain.Calculs.Deslletaments().

String domain.Manegador.MAvortaments (String *nom*, String *inici*, String *fi*) throws Exception

Calcul del nombre de parts abortats / dels parts totals; durant un periode i granja determinats.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 334 del fitxer Manegador.java.

Referències domain.Calculs.Avortaments(), domain.Manegador.c i domain.Manegador.nf.

String domain.Manegador.MRetiraments (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana del nombre de animalts retirats despres del part.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 350 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.Retiraments().

String domain.Manegador.MGarriDeslPerTruja (String *nom*, String *inici*, String *fi*) throws Exception

Calcul que indica el nombre de deslletaments / nombre de trujes productives.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 366 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.GarriDesIIPerTruja() i
domain.Manegador.nf.

String domain.Manegador.MIntervalEntreParts (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana entre els diferents parts d'un mateix animal.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 383 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.IntervalEntreParts() i
domain.Manegador.nf.

String domain.Manegador.MEdatDeslletament (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana d'edat dels garrins en realitzar el deslletament.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 399 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.EdatDeslletament() i
domain.Manegador.nf.

String domain.Manegador.MIDCF (String *nom*, String *inici*, String *fi*) throws Exception

IDCF:Interval mig entre el desllet (cicle n-1) i una cubricio fertil que ha donat part (cicle n).

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 415 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.IDCF() i domain.Manegador.nf.

String domain.Manegador.MIDOE (String *nom*, String *inici*, String *fi*) throws Exception

IDOE: Interval mig entre el desllet (cicle n-1) i la primera cubrició (cicle n).

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 432 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.IDOE() i domain.Manegador.nf.

String domain.Manegador.MNombrePrimeresGarrinades (String *nom*, String *inici*, String *fi*) throws Exception

Calcul del nombre mig de garrins nascuts vius durant el primer part de l'animal.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 449 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i
domain.Calculs.NombrePrimeresGarrinades().

String domain.Manegador.MEdatPrimerCobrimet (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana d'edat d'una truja al realitzar el primer cobrimet.

Tenint en compte una granja seleccionada i un període determinat

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 465 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.EdatPrimerCobriment() i
domain.Manegador.nf.

String domain.Manegador.MEdatPrimerPart (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana d'edat en el part de la truja d'una granja seleccionada y durant un període determinat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 481 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.EdatPrimerPart() i
domain.Manegador.nf.

String domain.Manegador.MEdatTrujesPart (String *nom*, String *inici*, String *fi*) throws Exception

Calcul de la mitjana de la edat de les trujes al part.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 497 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.EdatTrujesPart() i
domain.Manegador.nf.

String domain.Manegador.MNombreTrujesBaixa (String *nom*, String *inici*, String *fi*) throws Exception

Nombre de trujes diferents donades de baixa durant el periode indicat, de la granja seleccionada.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del periode que volem estudiar
fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 514 del fitxer Manegador.java.

Referències domain.Manegador.c i domain.Calculs.NombreTrujesBaixa().

String domain.Manegador.MNombreGarriPerTrujesBaixa (String *nom*, String *inici*, String *fi*) throws Exception

NombreGarriPerTrujesBaixa: garrins produïts en la seva vida les trujes de baixa dividit pel nombre de truges de baixa.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del periode que volem estudiar
fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 529 del fitxer Manegador.java.

String domain.Manegador.MEdatTrujesBaixa (String *nom*, String *inici*, String *fi*) throws Exception

Determina mitjana de la edat de les trujes al donar-les de baixa.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar
inici Indica el inici del periode que volem estudiar
fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 545 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.EdatTrujesBaixa i domain.Manegador.nf.

String domain.Manegador.IDBB (String *nom*, String *inici*, String *fi*) throws Exception

IDBB: Interval mig des del desllet (quan es l'últim moviment registrat) a la eliminació de la truja.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 562 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Calculs.IDBB() i domain.Manegador.nf.

String domain.Manegador.MNombreTrujes (String *nom*, String *inici*, String *fi*) throws Exception

Calcula el nun de femelles de una granja concreta y que no hagi estat donada de baixa fora del període seleccionat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 579 del fitxer Manegador.java.

Referències domain.Manegador.c i domain.Calculs.NombreTrujes().

String domain.Manegador.MNombreBerrosPresents (String *nom*, String *inici*, String *fi*) throws Exception

Calcula tots els animals mascles de la granja indicada i que encara no s'han donat de baixa o si se n'ha donat ha estat durant el període seleccionat.

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

inici Indica el inici del període que volem estudiar

fi Indica el fi del període que volem estudiar

Retorna:

El resultat de l'operació

Excepcions:

Definició a la línia 594 del fitxer Manegador.java.

Referències domain.Manegador.c i domain.Calculs.NombreBerrosPresentes().

String domain.Manegador.MTrujesPerBerro (String *nom*, String *inici*, String *fi*) throws Exception

Paràmetres:

nom Indica el identificador de la granja que es vol analitzar

Paràmetres:

inici Indica el inici del periode que volem estudiar

fi Indica el fi del periode que volem estudiar

Retorna:

El resultat de l'operacio

Excepcions:

Definició a la línia 606 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.TrujesPerBerro().

String domain.Manegador.stdIDOE ()

Calcul de la desviacio tipica IDOE, realitzant en el seu calcul principal.

Retorna:

El resultat de la std

Definició a la línia 624 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdIDOE.

String domain.Manegador.stdIDBB ()

Calcul de la desviacio tipica IDBB, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviacio tipica

Definició a la línia 636 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdIDBB.

String domain.Manegador.stdIDCF ()

Calcul de la desviacio tipica IDCF, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviacio tipica

Definició a la línia 649 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdIDCF.

String domain.Manegador.stdETB ()

Calcul de la desviacio tipica ETB, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 662 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdETB.

String domain.Manegador.stdETP ()

Calcul de la desviació típica ETP, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 675 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdETP.

String domain.Manegador.stdEPP ()

Calcul de la desviació típica EPP, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 688 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdEPP.

String domain.Manegador.stdGarrinDesll ()

Calcul de la desviació típica dels garrins deslletats, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 701 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdGarrinDesll().

String domain.Manegador.stdE1C ()

Calcul de la desviació típica E1C, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 714 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdE1C.

String domain.Manegador.stdNPG ()

Calcul de la desviació típica NPG, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 726 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdNPG.

String domain.Manegador.stdED ()

Calcul de la desviació típica ED, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica

Definició a la línia 739 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdED.

String domain.Manegador.stdNL ()

Calcul de la desviació típica NL, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica

Definició a la línia 752 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdNL().

String domain.Manegador.stdPTPdate ()

Calcul de la desviació típica PTPdate, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica

Definició a la línia 764 del fitxer Manegador.java.

Referències domain.Manegador.c i domain.Calculs.stdPTPdate().

String domain.Manegador.stdPTPROD ()

Calcul de la desviació típica PTPROD, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica

Definició a la línia 776 del fitxer Manegador.java.

Referències domain.Manegador.c i domain.Calculs.stdPTPROD().

String domain.Manegador.stdVIUS ()

Calcul de la desviació típica vius, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica

Definició a la línia 789 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdVIUS().

String domain.Manegador.stdMorts ()

Calcul de la desviació típica morts, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica

Definició a la línia 802 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdMorts.

String domain.Manegador.stdRetirats ()

Calcul de la desviació típica Retirats, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica

Definició a la línia 815 del fitxer Manegador.java.

Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdRetirats().

String domain.Manegador.stdleP ()

Calcul de la desviació típica leP, realitzant en el seu calcul principal.

Retorna:

El resultat de la desviació típica
Definició a la línia 827 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.stdleP.

String domain.Manegador.VariancaGarrinDesII ()

Calcul de la variància dels garrins deslletats, realitzant en el seu calcul principal.

Retorna:

El resultat de la variància
Definició a la línia 846 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaGarrinDesII().

String domain.Manegador.VariancaIDOE ()

Calcul de la variància IDOE, realitzant en el seu calcul principal.

Retorna:

El resultat de la variància
Definició a la línia 856 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaIDOE().

String domain.Manegador.VariancaIDBB ()

Calcul de la variància IDBB, realitzant en el seu calcul principal.

Retorna:

El resultat de la variància
Definició a la línia 866 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaIDBB().

String domain.Manegador.VariancaIDCF ()

Calcul de la variància IDCF, realitzant en el seu calcul principal.

Retorna:

El resultat de la variància
Definició a la línia 876 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaIDCF().

String domain.Manegador.VariancaETB ()

Calcul de la variància ETB, realitzant en el seu calcul principal.

Retorna:

El resultat de la variància
Definició a la línia 886 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaETB().

String domain.Manegador.VariancaETP ()

Calcul de la variancia ETP, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 896 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaETP().

String domain.Manegador.VariancaEPP ()

Calcul de la variancia EPP, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 906 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaEPP().

String domain.Manegador.VariancaE1C ()

Calcul de la variancia E1C, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 916 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaE1C().

String domain.Manegador.VariancaNPG ()

Calcul de la variancia NPG, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 926 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaNPG().

String domain.Manegador.VariancaED ()

Calcul de la variancia ED, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 936 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaED().

String domain.Manegador.VariancaNL ()

Calcul de la variancia NL, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 946 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaNL().

String domain.Manegador.VariancaPTPdate ()

Calcul de la variancia PTPdate, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 956 del fitxer Manegador.java.
Referències domain.Manegador.c i domain.Calculs.VariancaPTPdate().

String domain.Manegador.VariancaPTPROD ()

Calcul de la variancia PTPROD, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 965 del fitxer Manegador.java.
Referències domain.Manegador.c i domain.Calculs.VariancaPTPROD().

String domain.Manegador.VariancaVIUS ()

Calcul de la variancia vius, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 974 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaVIUS().

String domain.Manegador.VariancaMorts ()

Calcul de la variancia morts, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 984 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaMorts().

String domain.Manegador.VariancaRetirats ()

Calcul de la variancia Retirats, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 994 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaRetirats().

String domain.Manegador.VariancaleP ()

Calcul de la variancia leP, realitzant en el seu calcul principal.

Retorna:

El resultat de la variancia
Definició a la línia 1004 del fitxer Manegador.java.
Referències domain.Manegador.c, domain.Manegador.nf i domain.Calculs.VariancaleP().

List domain.Manegador.MViusSetmanaGrafic (String nom, String inici, String fi) throws Exception

Definició a la línia 1018 del fitxer Manegador.java.

Referències domain.Manegador.cg i domain.CalculsGrafics.ViusSetmanaGrafic().

List domain.Manegador.MMortsSetmanaGrafic (String nom, String inici, String fi) throws Exception

Definició a la línia 1022 del fitxer Manegador.java.

Referències domain.Manegador.cg i domain.CalculsGrafics.MortsSetmanaGrafic().

List domain.Manegador.MRetiradosSetmanaGrafic (String nom, String inici, String fi) throws Exception

Definició a la línia 1026 del fitxer Manegador.java.

Referències domain.Manegador.cg i domain.CalculsGrafics.RetiramentsSetmanaGrafic().

List domain.Manegador.MNumPartsSetmanaGrafic (String nom, String inici, String fi) throws Exception

Definició a la línia 1030 del fitxer Manegador.java.

Referències domain.Manegador.cg i domain.CalculsGrafics.NumPartsSetmanaGrafic().

List domain.Manegador.MNDeslletamentsSetmanaGrafic (String nom, String inici, String fi) throws Exception

Definició a la línia 1034 del fitxer Manegador.java.

Referències domain.Manegador.cg i domain.CalculsGrafics.NDeslletamentsSetmanaGrafic().

List domain.Manegador.MAdoptatsSetmanaGrafic (String nom, String inici, String fi) throws Exception

Definició a la línia 1038 del fitxer Manegador.java.

Referències domain.CalculsGrafics.AdoptatsSetmanaGrafic() i domain.Manegador.cg.

List domain.Manegador.MNCubSetmanaGrafic (String nom, String inici, String fi) throws Exception

Definició a la línia 1042 del fitxer Manegador.java.

Referències domain.Manegador.cg i domain.CalculsGrafics.NCubSetmanaGrafic().

List domain.Manegador.MNMortsDiaGrafic (String nom, String inici, String fi) throws Exception

Definició a la línia 1049 del fitxer Manegador.java.

Referències domain.Manegador.cg i domain.CalculsGrafics.NMortsDiaGrafic().

List domain.Manegador.MNViusDiaGrafic (String nom, String inici, String fi) throws Exception

Definició a la línia 1053 del fitxer Manegador.java.

Referències domain.Manegador.cg i domain.CalculsGrafics.NViusDiaGrafic().

List domain.Manegador.MNPartsDiaGrafic (String nom, String inici, String fi) throws Exception

Definició a la línia 1057 del fitxer Manegador.java.

Referències `domain.Manegador.cg` i `domain.CalculsGrafics.NPartsDiaGrafic()`.

List `domain.Manegador.MNDeslletamentsDiaGrafic (String nom, String inici, String fi)` throws Exception

Definició a la línia 1061 del fitxer `Manegador.java`.

Referències `domain.Manegador.cg` i `domain.CalculsGrafics.NDeslletamentsDiaGrafic()`.

List `domain.Manegador.MNViusCicleGrafic (String nom, String inici, String fi)` throws Exception

Definició a la línia 1068 del fitxer `Manegador.java`.

Referències `domain.Manegador.cg` i `domain.CalculsGrafics.NViusCicleGrafic()`.

List `domain.Manegador.MNAnimalsCicleGrafic (String nom, String inici, String fi)` throws Exception

Definició a la línia 1072 del fitxer `Manegador.java`.

Referències `domain.Manegador.cg` i `domain.CalculsGrafics.NAnimalsCicleGrafic()`.

List `domain.Manegador.MNDeslletatsCicleGrafic (String nom, String inici, String fi)` throws Exception

Definició a la línia 1076 del fitxer `Manegador.java`.

Referències `domain.Manegador.cg` i `domain.CalculsGrafics.NDeslletatsCicleGrafic()`.

List `domain.Manegador.MIntervalDeslletamentGrafic (String nom, String inici, String fi)` throws Exception

Definició a la línia 1084 del fitxer `Manegador.java`.

Referències `domain.Manegador.cg` i `domain.CalculsGrafics.IntervalDeslletamentGrafic()`.

List `domain.Manegador.MIntervalIDCFGrafic (String nom, String inici, String fi)` throws Exception

Definició a la línia 1088 del fitxer `Manegador.java`.

Referències `domain.Manegador.cg` i `domain.CalculsGrafics.IntervalIDCFGrafic()`.

Referència de la Classe `domain.Operacions`

Descripció Detallada

Classe que s'encarrega de proporcionar mètodes per fer consultes SQL i obtenir els resultats.

Autor:

Admin

Definició a la línia 23 del fitxer `Operacions.java`.

Mètodes públics

- [**Operacions**](#) (**Connection [conn](#)**) throws Exception
Creates a new instance of [Operacions](#).
- void [**CreaConsulta**](#) (String s, String f, String w)
Mètode per crear una consulta SQL.
- void [**AfegeixWhere**](#) (String w)

Metode per afegir mes condicions al Where d'una consulta ja creada.

- **List< String > [ObteLlistaQuery \(\)](#)**
Metode per obtenir la consulta en una llista de String.
- **ResultSet [ExeConsultaGenerica](#) (String database, List< String > llista_consulta) throws SQLException**
Metode que executa la consulta SQL.
- **List< String > [CreaLlistaResultats](#) (ResultSet rs, String columna) throws SQLException**
Retorna a partir d'un ResultSet una llista que conte les dades del ResultSet a la Columna indicada.
- **void [TancarOperacio](#) (ResultSet rs) throws SQLException**
Metode encarregat de tancar una consulta un cop finalitzada.

Atributs Privats

- Connection [conn](#) = null
- Statement [selectStmt](#)
- List< String > [llistaquery](#)
- List< String > [llistaresultat](#)

Documentació del Constructor i el Destructor

domain.Operacions.Operacions (Connection conn) throws Exception

Creates a new instance of [Operacions](#).

Definició a la línia 32 del fitxer Operacions.java.

Referències domain.Operacions.conn.

Documentació de les Funcions Membre

void domain.Operacions.CreaConsulta (String s, String f, String w)

Metode per crear una consulta SQL.

Paràmetres:

s String que correspon al Select de la consulta SQL

f String que correspon al From de la consulta SQL

w String que correspon al Where de la consulta SQL

Definició a la línia 44 del fitxer Operacions.java.

void domain.Operacions.AfegeixWhere (String w)

Metode per afegir mes condicions al Where d'una consulta ja creada.

Paràmetres:

w String que correspon al Where que es vol afegir de la consulta SQL

Definició a la línia 58 del fitxer Operacions.java.

List<String> domain.Operacions.ObteLlistaQuery ()

Metode per obtenir la consulta en una llista de String.

Retorna:

List<String> que conte el la consulta SQL
Definició a la línia 68 del fitxer Operacions.java.

ResultSet domain.Operacions.ExeConsultaGenerica (String *database*, List< String > *llista_consulta*) throws SQLException

Metode que executa la consulta SQL.

Paràmetres:

database String que aporta el nom de la Base de Dades sobre la qual es fara la consulta
llista_consulta List<String> que conte el la consulta SQL

Excepcions:

java.sql.SQLException

Retorna:

ResultSet que conte el resultat de la consulta SQL
Definició a la línia 83 del fitxer Operacions.java.

List<String> domain.Operacions.CreaLlistaResultats (ResultSet *rs*, String *columna*) throws SQLException

Retorna a partir d'un ResultSet una llista que conte les dades del ResultSet a la Columna indicada.

Paràmetres:

rs ResultSet que conte el resultat d'una consulta SQL
columna String on s'indica el nom de la columna de resultats que volem introduir a la llista

Excepcions:

java.sql.SQLException

Retorna:

List<String> llista amb les dades del ResultSet a la Columna indicada
Definició a la línia 113 del fitxer Operacions.java.

void domain.Operacions.TancarOperacio (ResultSet *rs*) throws SQLException

Metode encarregat de tancar una consulta un cop finalitzada.

Paràmetres:

rs ResultSet que conte el resultat d'una consulta SQL

Excepcions:

java.sql.SQLException
Definició a la línia 139 del fitxer Operacions.java.

Annex C

IDE Netbeans

Netbeans en aquest cas ha estat l'entorn de desenvolupament que hem elegit per a la implementació del software d'aquest projecte.

NetBeans es refereix a una plataforma per al desenvolupament d'aplicacions d'escriptori usant Java i un entorn de desenvolupament integrat (IDE) desenvolupat usant la Plataforma NetBeans.

La plataforma NetBeans permet que les aplicacions siguin desenvolupades a partir d'un conjunt de components de programari anomenats mòduls. Un mòdul és un fitxer Java que conté classes de Java escrites per a interactuar amb les APIs de NetBeans i un arxiu especial (manifest file) que ho identifica com mòdul.

Les aplicacions construïdes a partir de mòduls poden ser esteses agregant-li nous mòduls. Degut al fet que els mòduls poden ser desenvolupats independentment, les aplicacions basades en la plataforma NetBeans poden ser esteses fàcilment per altres desenvolupadors de programari.

NetBeans és un projecte de codi obert de gran èxit amb una gran base d'usuaris, una comunitat en constant creixement. Sun Microsystems va fundar el projecte de codi obert NetBeans al juny 2000 i continua sent el patrocinador principal dels projectes.



Annex D

Llicència GPL

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps:

(1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents.

States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used

unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation

or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the

compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions.

Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-

peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS**How to Apply These Terms to Your New Programs**

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>
```

```
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
```

```
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary.

For more information on this, and how to apply and follow the GNU GPL, see <<http://www.gnu.org/licenses/>>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

But first, please read <<http://www.gnu.org/philosophy/why-not-lgpl.html>>.